

Transformation von CIMOSA-Modellen in Petrinetz-Modelle mit PACE

Bernd Eichenauer, IBE Ingenieurbüro (Eichenauer@ibepace.com)
Martin Zelm, CIMOSA Association e.V. (Zelm@cimosade.de)

Vorbemerkung

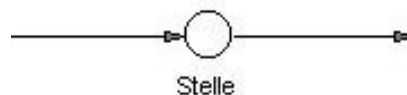
CIMOSA (CIM Open System Architecture) verfolgt den Ansatz zu einer prozessorientierten, ganzheitlichen Unternehmensmodellierung und Integration [CIMOSA-93,-96,-99, Vernadat-96]. Das Konzept wurde von dem Konsortium AMICE in einer Reihe von Projekten im europäischen Forschungsprogramm entwickelt und validiert. CIMOSA liefert Anforderungsmodelle von diskreten Prozessen -insbesondere von Geschäftsmodellen- in einer für den Benutzer transparenten Darstellung. Mit dem Werkzeug PACE [PACE-99] erstellte Petrinetz-Modelle zielen auf die Modellierung der gleichen Sachverhalte mit dem Ziel, möglichst exakte ablauffähige Modelle zu erreichen, die quantitative Aussagen ermöglichen. Beide Modellierungsmethoden behandeln die gleichen Sachverhalte mit verschiedenen Zielsetzungen, wobei CIMOSA-Modelle mehr darauf zielen, die Sachverhalte, z.B. für eine Entscheidungsunterstützung, qualitativ verständlich zu machen. PACE-Modelle liefern mathematisch exakte Simulationsergebnisse, die Modelle sind aber im allgemeinen wegen der für die Simulation erforderlichen formalen Darstellung weniger transparent.

Im folgenden werden Regeln für die Implementierung bzw. die Transformation von CIMOSA-Modellen in PACE-Modelle vorgeschlagen und danach auf ein einfaches Beispiel angewandt. Diese Regeln müssen später noch an weiteren komplexeren Beispielen erprobt und durch weitere Regeln ergänzt werden, welche einen häufig auftretenden Satz von Ereignissen und Aktivitäten einschließen.

Allgemeine Transformationsregeln

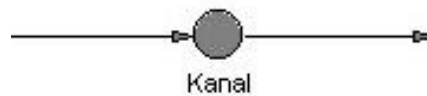
Wir legen folgendes Umsetzungsverfahren fest:

1. Enterprise Activities werden in PACE-Module umgesetzt. Für bestimmte standardisierte Enterprise Activities werden Standard-Module in einer PACE-Modulbibliothek bereitgestellt.
2. Ereignisse werden in PACE-Module abgebildet. Für bestimmte standardisierte Ereignisse (z.B. Zeit-Ereignisse) werden Module in einer PACE-Modulbibliothek vorgesehen.
3. Einfache Verbindungen zwischen CIMOSA-Bausteinen (Ereignisse, Enterprise Activities) werden in das PACE-Konstrukt:



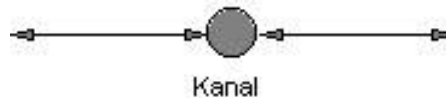
umgesetzt.

4. Mehrfache Verbindungen zwischen CIMOSA-Bausteinen in einer Richtung werden durch das Konstrukt:



dargestellt.

5. Treten Verbindungen in beiden Richtungen auf, so ist das Konstrukt :



zu verwenden. In Spezialfällen kann der Kanal auch in eine Stelle degenerieren.

Beispiel:

Wir wenden die Transformationsregeln auf das folgende Beispiel einer Waschstraße an, die nach der CIMOSA-Methode modelliert ist :

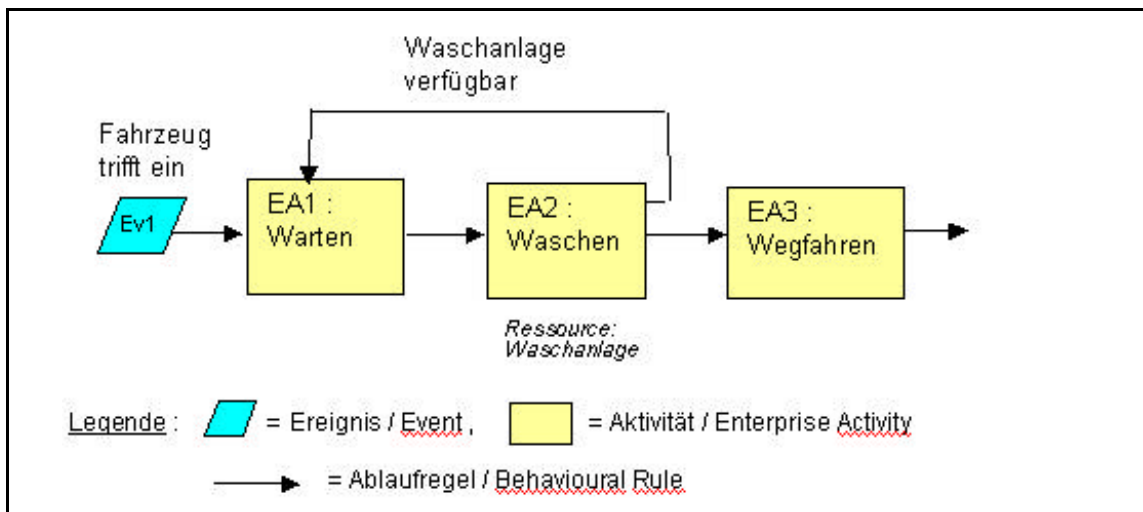


Abbildung 1a: Anwendungsbeispiel einer Waschstraße mit CIMOSA modelliert

und erhalten folgendes Hauptnetz des zugeordneten PACE-Modells:

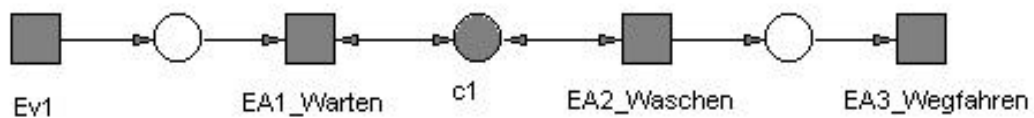


Abbildung 1b: Dieselbe Waschstraße modelliert als Petrinetz mit PACE

Modellierung von Ereignissen

Es gibt in PACE zahlreiche Möglichkeiten zur Modellierung von Ereignissen. Welche Methode jeweils eingesetzt wird, hängt sowohl von der Zielsetzung des Modellierers als auch von der Art des Ereignisses ab. Eine endgültige Anleitung der Umsetzung von CIMOSA nach PACE muß die in CIMOSA möglichen Ereignisse klassifizieren und deren Umsetzung beschreiben und/oder Standard-Module aus einer PACE-Bibliothek für sie vorgeben.

Wir betrachten die Umsetzung des im oben genannten Beispiel einer Waschanlage auftretenden Ereignisses:

Ereignis / Event

Ereignis (Event)	
Type	Material (Ankunft)
Identifier	EV1
Name	Fahrzeug trifft ein
Description	Fahrzeug trifft an der Waschanlage ein
Triggers	Enterprise Activity 'Fahrzeug warten'
Resource	Fahrer
Object	Fahrzeug
Häufigkeit	Exponentialverteilung, Mean = 12

Es handelt sich dabei um eine Folge exponentiell verteilter Zeitpunkte, zu denen jeweils ein Fahrzeug an der Waschanlage vorfährt. Das geht aus der CIMOSA-Anweisung:

WHEN (START WITH Event1)
Fahrzeug trifft ein DO (EA1) Warten

hervor.

Von den Angaben in der Tabelle benötigen wir nur die Object-Angabe und die Angabe der Häufigkeit. Die übrigen Angaben, soweit sie hier interessieren, sind schon in unserem Hauptnetz berücksichtigt worden. Z.B. wird die Angabe „Triggers“ schon mit der Verbindung von Ev1 nach EA1_Warten berücksichtigt.

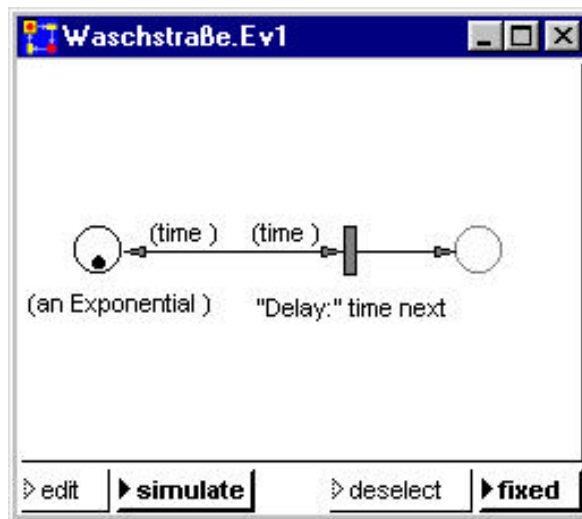
Die Angabe des „Object“ kann später dazu verwendet werden, um die Standard-Ikone für Marken durch Abbildungen von Fahrzeugen zu ersetzen. und damit das PACE-Modell auch visuell an den modellierten Sachverhalt anzupassen.¹

Im folgenden werden für die Erzeugung von Ereignissen, die zu exponentiell verteilten Zeitpunkten stattfinden, zwei Module angegeben.

Fester Mittelwert

Eine Exponentialverteilung ist durch die Angabe ihres Mittelwerts vollständig bestimmt. Wenn der Mittelwert fest vorgegeben ist, kann der folgende Modul verwendet werden:

¹ Auf Maßnahmen zur „Verschönerung“ von PACE-Modellen wird hier nicht eingegangen. Sie sind in der PACE-Dokumentation ausführlich beschrieben.



In die linke Stelle wird eine sog. Initialmarke gelegt, welche eine Methode zur Berechnung von Werten der Exponentialverteilung trägt. Beim Ablauf des Modells wird diese Berechnungsmethode zu der rechts von der Stelle liegenden Transition gebracht, wo mit der Methode „time next“ der nächste Wert der Verteilung bestimmt wird. Dieser Wert gibt die Wartezeit bis zur Ankunft des nächsten Fahrzeugs an. Die Berechnungsvorschrift wird vorher über das Menü für Initialmarken festgelegt:

tokens	attributes	code
1	1	Exponential mean: 12

Wenn diese Zeit verstrichen ist, läuft die Marke mit der Berechnungsvorschrift von der linken Stelle zur Transition, wird dort ausgewertet und danach zur linken Stelle zurückgebracht. Gleichzeitig wird eine Marke (ein Fahrzeug) in die rechts liegende Stelle gebracht. Diese Stelle ist schwächer gezeichnet, weil sie nicht in diesem Modul, sondern im Hauptnetz (eine Ebene höher, siehe weiter oben) vereinbart ist.

Veränderlicher Mittelwert

Der unter 1. beschriebene Modul hat den Nachteil, daß die Berechnungsvorschrift für die nächste Wartezeit durch manuelle Eingabe fest vorgegeben wird. Diese Vorschrift kann nur durch manuelle Eingabe einer neuen Verteilung bzw. eines neuen Mittelwerts verändert werden.

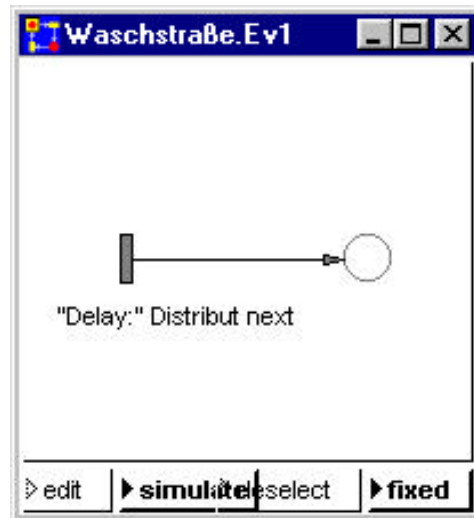
Gelegentlich ändert sich die Exponentialverteilung infolge äußerer Ereignisse oder zeitabhängig und man möchte die Verteilung während des Ablaufs der Simulation verändern.

In diesem Fall ist es zweckmäßig, die Berechnungsvorschrift beim Initialisieren des Modells in einer Netzvariablen oder einer globalen Variablen zu speichern. Soll die Verteilung geändert werden, so wird in der Variablen einfach die neue Verteilung gespeichert.

Fügen wir in den Initialisierungscode von PACE beispielsweise die Zuweisung:

Distribut := Exponential mean: 12.

ein, wobei „Distribut“ eine globale Variable ist, so würde der Modul „Ev1“ wie folgt aussehen:



Die beiden angegebenen Module könnten als Standard-Module in einer PACE-Bibliothek gespeichert und fallweise in Modelle eingesetzt und adaptiert werden.

Modellierung von Kanälen

Kanäle als Schnittstelle zwischen Moduln enthalten nur passive Netzelemente, d.h. Stellen oder Kanäle. Für jede einzelne Verbindung zwischen zwei Moduln ist eine Stelle vorzusehen. Sollen mehrere (Unter-)Verbindungen zusammengefaßt werden, so kann der Kanal weitere (Unter-)Kanäle enthalten.

Im vorliegenden Beispiel öffnen wir den Modul „c1“ im Hauptnetz und fügen gemäß der CIMOSA-Beschreibung (siehe Seite 2) die zwei Stellen für das „Triggern“ des Waschens und für die Rückmeldung „Waschanlage verfügbar“ ein:



Modellierung von Aktivitäten

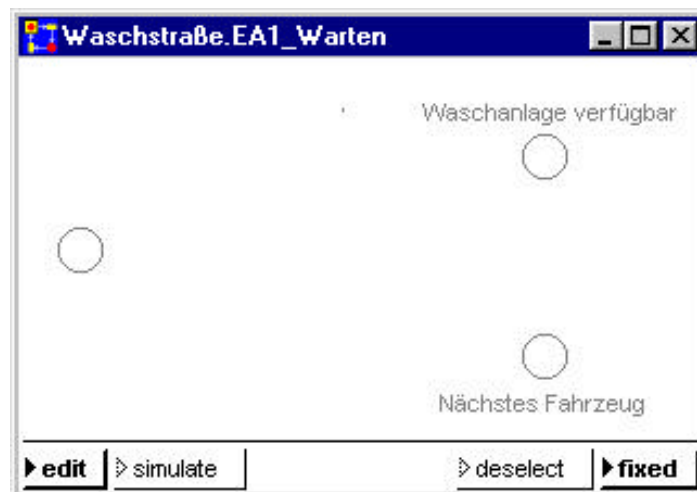
In unserem Beispiel kommen die drei CIMOSA-Aktivitätstypen „Lagern“, „Transformieren“ und „Bewegen“ vor.² Im folgenden werden die im Hauptnetz angegebenen drei Enterprise Activities modelliert.

Warten (Aktivitätstyp: Lagern)

Wir öffnen den Modul EA1 und erhalten folgendes Bild:



Links haben wir die Schnittstelle zum Ereignismodul Ev1, rechts die zur Enterprise Activity EA2. Um die Verbindungen im Detail herstellen zu können, setzen wir zunächst die Verfeinerung des Kanals ein und kommen zu folgendem Modul:

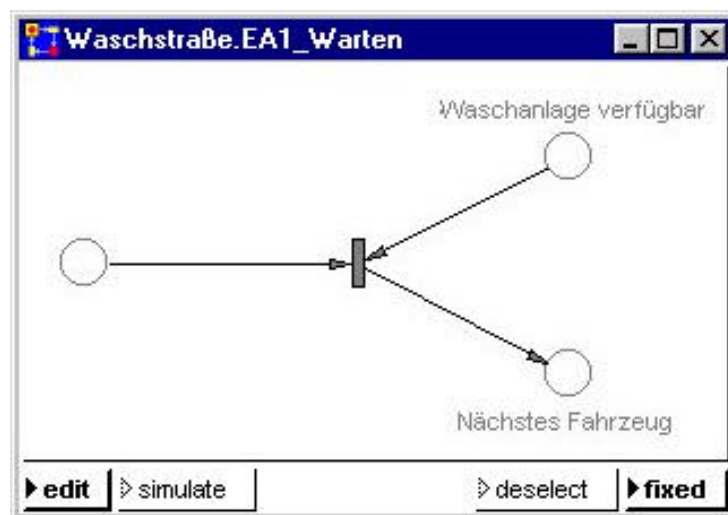


² Auf dem niedrigeren Niveau von PACE ist diese problem-orientierte Unterscheidung nicht mehr sinnvoll, da aus der PACE-Darstellung der Aktivitätstyp nicht mehr ersichtlich ist.

Aus dem Hauptnetz und der CIMOSA-Beschreibung des Moduls „Warten“:

Aktivität (Enterprise Activity)	
Type	Lagern : Material oder Information
Identifizier	EA1
Name	Warten
Description	Fahrzeug wartet in vor der Waschanlage
Function Input	Fahrzeug
Control Input	Waschanlage Status
Resource Input	Fahrer
Function Output	Fahrzeug
Control Output	
Resource Output	Fahrer Wartezeit
Duration Min / Max / Average	Durchschnittl. Wartezeit
Ending Status	Fahrzeug freigegeben
-	

entnehmen wir, daß das Fahrzeug warten muß, bis die Waschanlage frei ist. Diese Bedingung kann erfüllt werden, indem wir die drei Stellen über eine Transition verknüpfen:



In der Tat kann die Transition nur schalten, wenn in der linken Stelle und in der Stelle „Waschanlage verfügbar“ je eine Marke liegt. Ist dies der Fall, dann wird in die Stelle „Nächstes Fahrzeug“ eine Marke gelegt, die das zu waschende Fahrzeug repräsentiert. Da die Waschanlage zu Beginn der Simulation frei ist, müssen wir in die Stelle „Waschanlage verfügbar“ eine Initialmarke legen. Damit ist die CIMOSA-Anweisung:

```

WHEN Ending Status Waschanlage frei
      AND      Fahrzeug verfügbar DO (EA2)   Waschen
  
```

erfüllt und wir haben damit auch das Muster eines Standard-Moduls gewonnen, der ein Objekt liefert, wenn zwei Bedingung erfüllt sind.

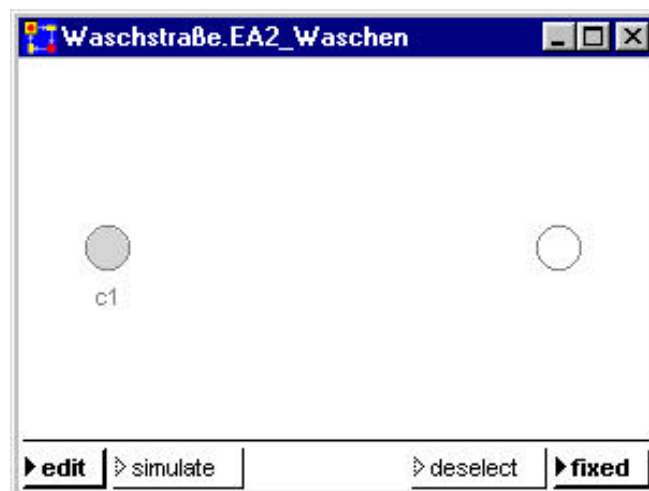
Erwähnenswert ist, daß sich das Bild des Kanals c1 nach Modellierung des Moduls „EA1_Warten“ automatisch wie folgt ergänzt hat:



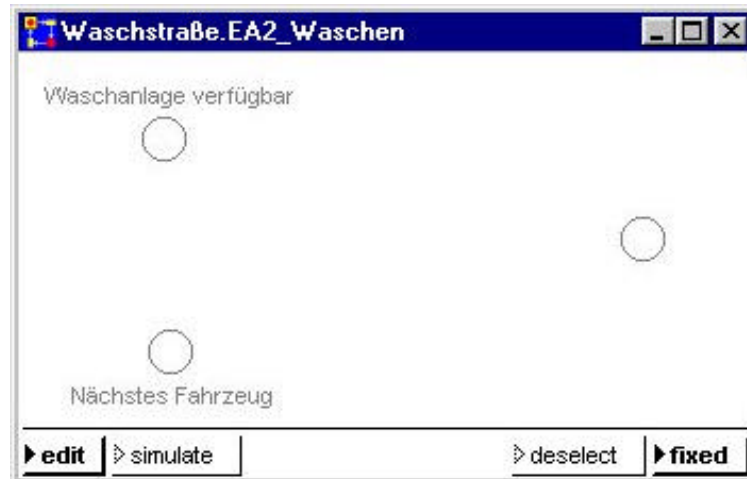
Die Kopplung der Kanalelemente mit der im Hauptnetz liegenden Enterprise Activity wird ersichtlich.

Waschen (Aktivitätstyp: Transformieren)

Wie bei EA1 öffnen wir auch hier zunächst den Modul EA2 und sehen folgendes Bild:



Es entspricht dem Bild, das wir zu Beginn der Modellierung von „EA1_Warten“ vorgefunden haben. Wir verfeinern auch hier wieder den Kanal und erhalten:



Aus der Beschreibung

Aktivität (Enterprise Activity)	
Type	Transformieren : Material oder Information
Identifizier	EA3
Name	Waschen
Description	Fahrzeug fährt in die Waschanlage und wird gewaschen, wobei die Waschanlage für weitere Fahrzeuge gesperrt wird
Function Input	Fahrzeug
Control Input	-
Resource Input	Waschanlage Fahrer
Function Output	Fahrzeug gewaschen
Control Output	-
Resource Output	Waschanlage Arbeitszeit Fahrer Wartezeit
Duration	Durschnittl. Waschzeit 10 min
Ending Status	Waschanlage frei

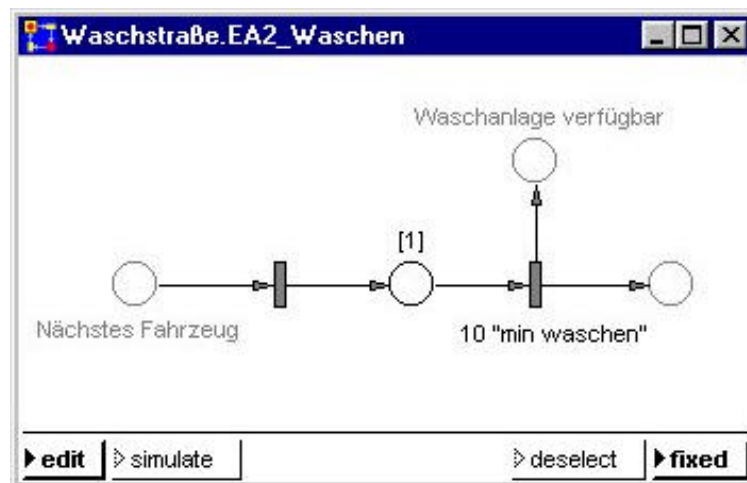
entnehmen wir zwei für die Modellierung wesentliche Tatbestände::

- Es darf sich jeweils nur eine Marke(ein Fahrzeug) in dem Modul „Waschen“ aufhalten.
- Ein Waschvorgang dauert im Schnitt 10 min.

Über das weitere Verfahren gibt die CIMOSA-Anweisung:

WHEN Ending Status Fahrzeug gewaschen
AND Waschanlage frei DO (EA3) Wegfahren

Auskunft. Ein Teilnetz, daß diese Forderungen realisiert, ist in der nächsten Abbildung gezeichnet:



Abgesehen von Positionsänderungen der Stellen, die als Schnittstellen nach außen fungieren (schwächer gezeichnet), wurden in den Modul drei weitere Netzelemente eingebracht. Die linke Transition überführt eine Marke in die mittlere Stelle (Waschplatz), die mit „[1]“ gekennzeichnet ist. „[1]“ bedeutet, daß nur jeweils eine Marke auf dieser Stelle liegen darf. Die rechte Transition sorgt dafür, daß die Marke den Waschplatz erst nach 10 Simulations-Zeiteinheiten (1 Einheit = 1 Minute) verlassen kann.

Sind 10 Zeiteinheiten vergangen, so schaltet die Transition und legt jeweils eine Marke in die beiden Ausgangsstellen. Die Marke in der Stelle „Waschanlage verfügbar“ ist eine Control-Ausgabe und wird wie früher erläutert im Modul „Warten“ ausgewertet. Die Marke in der rechten Stelle modelliert das Fahrzeug, welches den Waschplatz gerade verlassen hat.

Der Modul kann als Prototyp eines Bearbeitungsmoduls mit einem Bediener angesehen werden.

Zu erwähnen bleibt, daß sich durch die Modellierung von EA2 das Interface zu EA1 automatisch vervollständigt hat:

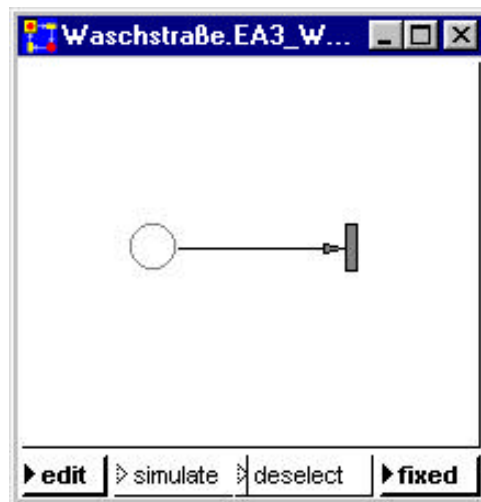


Wegfahren (Aktivitätstyp Bewegen)

Der letzte der drei Moduln ist ebenfalls sehr einfach. Öffnen wir den Modul EA3, so sehen wir das folgende Bild:



Es enthält nur die Schnittstelle zum Modul EA2. Wir fügen nur eine Transition ein und erhalten:

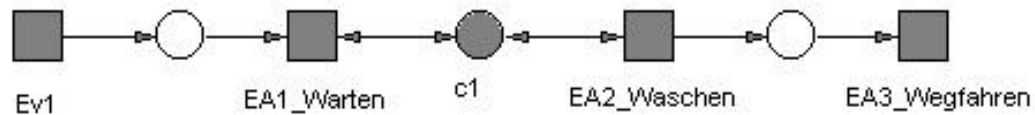


Die Transition schaltet, wenn sich in der Stelle eine Marke befindet und entfernt die Marke (das Fahrzeug) wieder aus dem Modell.

Ergebnisse und Diskussion

Wir stellen die entwickelten PACE-Beschreibungen nochmals zusammen:

Ebene 1:



Ebene 2:

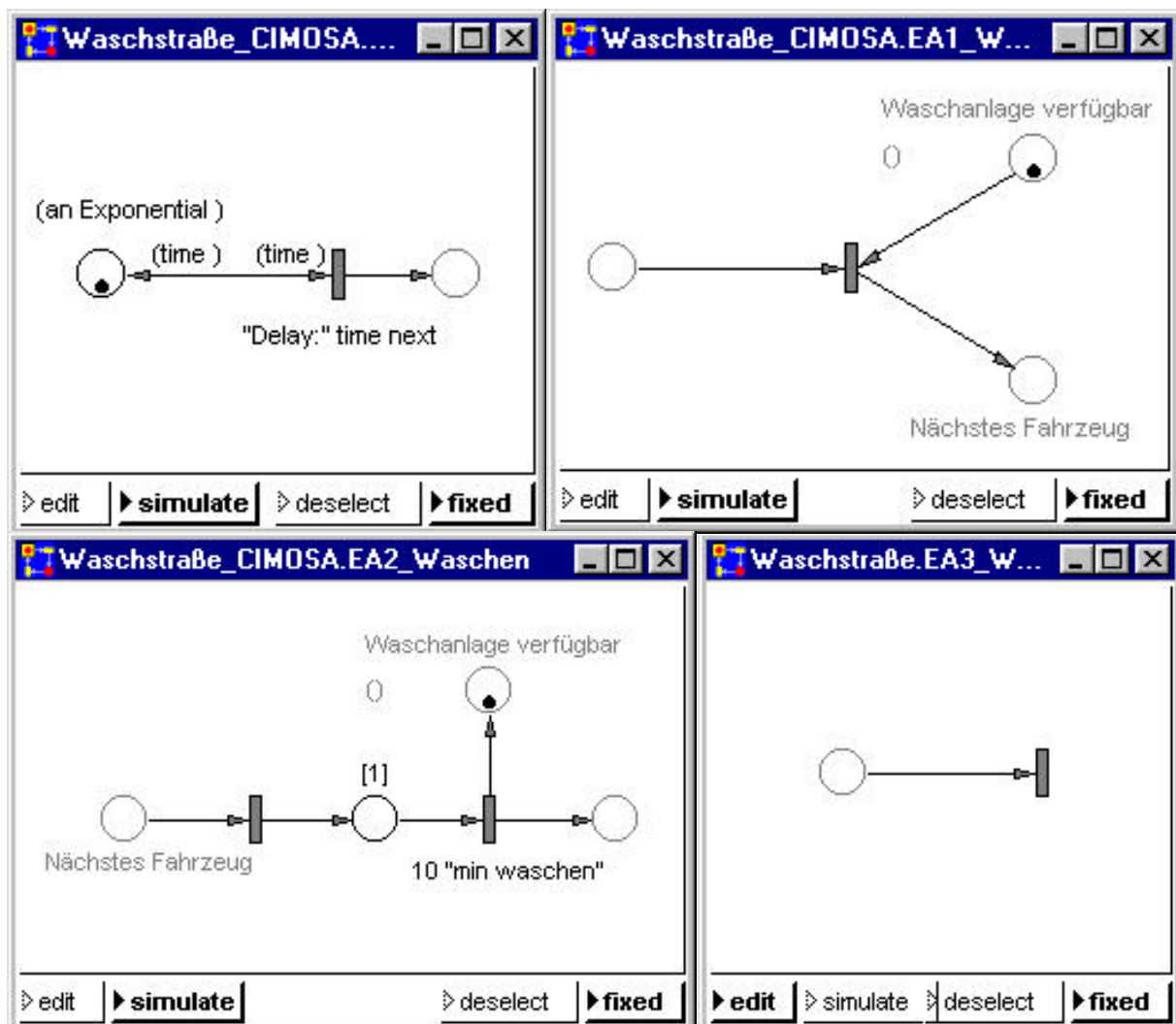


Abbildung 2: Waschstraße modelliert in zwei Petrinetz-Ebenen mit PACE

Um den Nutzen einer Transformation nach PACE und die dadurch möglichen Aussagen über die Dynamik eines Modells zu verdeutlichen, wollen wir untersuchen, wie die Warteschlange vor der Waschanlage aussieht.

Kommen mehr Fahrzeuge, als sofort abgearbeitet werden können, so sammeln sich die Fahrzeuge in der am weitesten links angegebenen Stelle in Ebene 1. Diese Stelle ist auch in der rechten bzw. linken Stelle von Abbildung „Waschstraße.Ev1“ bzw. „Waschstraße.EA1_Warten“ dargestellt.

PACE bietet standardmäßig die Möglichkeit, an eine Stelle ein Histogramm oder ein Liniendiagramm anzuschließen. Lassen wir das Modell ablaufen, so erhalten wir:

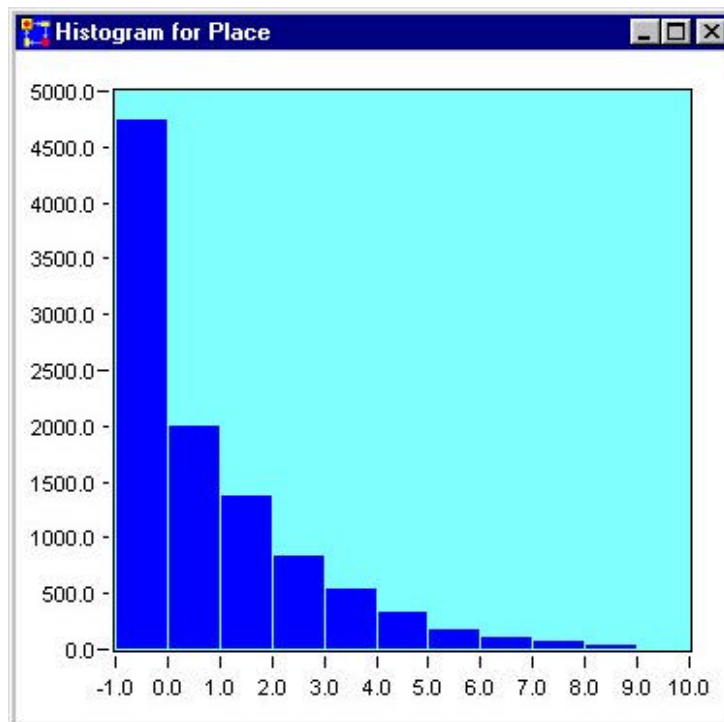


Abbildung 3: Häufigkeitsverteilung Anzahl wartender Fahrzeuge

Das Histogramm lässt erkennen, daß praktisch nie mehr als 6 Fahrzeuge vor der Waschanlage warten und daß man in ca. 40% alle Fälle sofort in die Waschstraße einfahren kann.

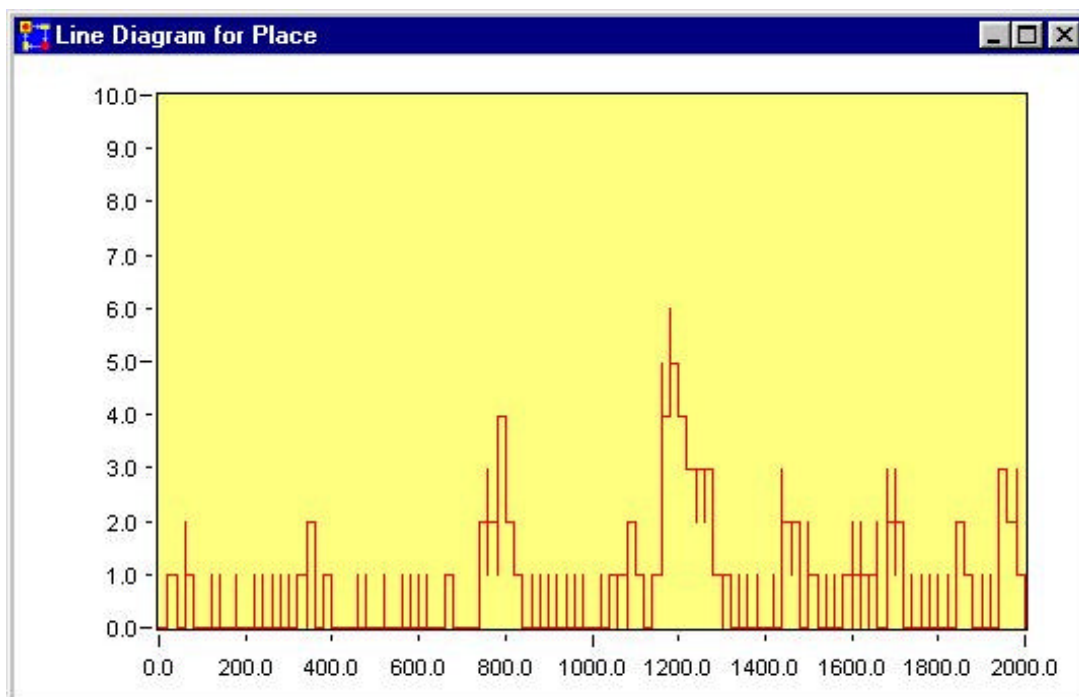


Abbildung 4: Belegung der Waschstraße mit Fahrzeugen

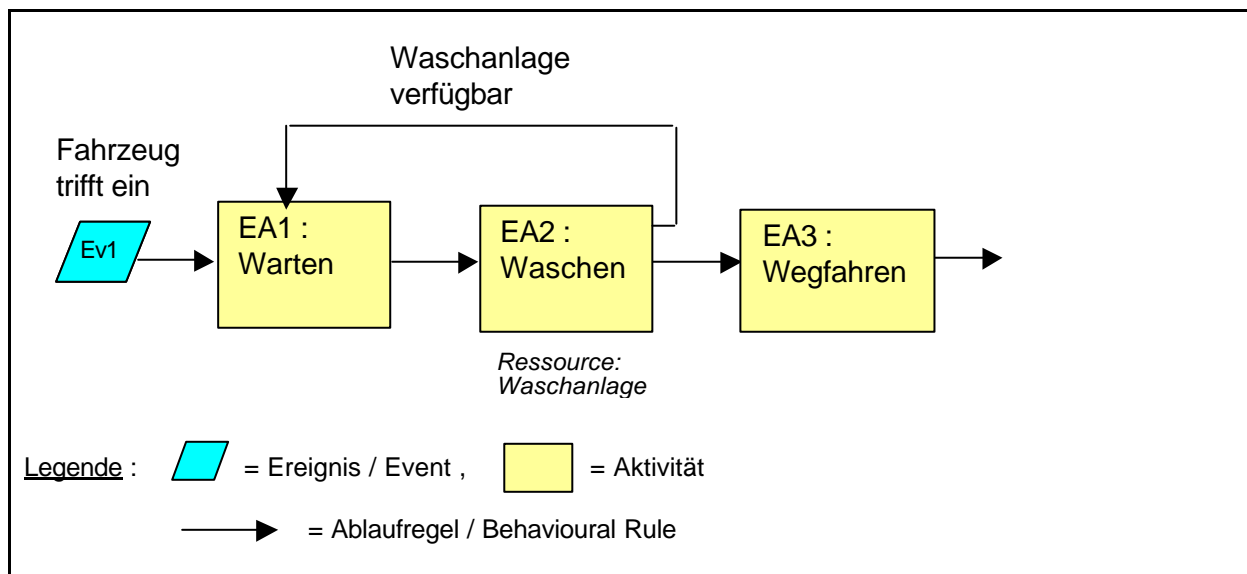
Das Liniendiagramm zeigt eine mit den Modellannahmen verträgliche zufällige Belegung der Waschstraße. Die maximale Belegung beträgt dabei sechs Fahrzeuge.

Das Beispiel zeigt die Möglichkeiten, Szenarien von Geschäftsprozessen mit High Level Petrinetzen zu simulieren und Vorhersagen durchzuführen. Der Vorteil des dargestellten zweistufigen Ansatzes besteht darin, daß die Anforderungen – hinsichtlich der Prozesse, Aktivitäten, Ressourcen und Organisation – in einer Benutzer orientierten Weise definiert werden. Die folgende Simulation mit Petrinetzen liefert dann die dazugehörigen mathematisch exakten Resultate.

Weiterführende Arbeiten sind erforderlich, um mit der beschriebenen Transformationsmethode komplexe Szenarien von Ereignisgesteuerten Prozeßketten (EPKs) zu modellieren und zu simulieren. Dafür müssen zunächst die allgemeinen Transformationsregeln erweitert werden, um komplexe Szenarien abbilden zu können. Ferner müssen wiederverwendbare Bausteine von Ereignistypen und Aktivitätstypen oder Bibliotheken von Teilmodellen in CIMOSA spezifiziert und die entsprechenden Sub-Netze im Werkzeug PACE definiert werden. PACE stellt die Methoden zur Verfügung, um derartige Modellteile mit wirtschaftlich vertretbarem Aufwand zusammensetzen. Damit wird ein allgemeiner ingenieurmäßiger Ansatz zur Transformation von CIMOSA Modellen in Petrinetz-Modelle mit PACE möglich.

Anhang: Modellierungsbeispiel Waschstraße

In dem folgenden Anhang ist das CIMOSA-Modell, also die Abbildung der Waschanlage und die Templates der Modellierungskonstrukte zusammengefaßt.



Modell der Waschstraße

Templates der Aktivitäten und Ereignisse**Aktivitätstyp 'Lagern'***Beispiel: Fahrzeug wartet*

Aktivität (Enterprise Activity)	
Type	Lagern : Material oder Information
Identifizier	<i>EAI</i>
Name	Warten
Description	Fahrzeug wartet in vor der Waschanlage
Function Input	Fahrzeug
Control Input	Waschanlage Status
Resource Input	Fahrer
Function Output	Fahrzeug
Control Output	
Resource Output	Fahrer Wartezeit
Duration Min / Max / Average	Durchschnittl. Wartezeit
Ending Status	Fahrzeug freigegeben
-	

Aktivitätstyp 'Transformieren'*Beispiel :Fahrzeug waschen*

Aktivität (Enterprise Activity)	
Type	Transformieren : Material oder Information
Identifizier	<i>EA3</i>
Name	Waschen
Description	Fahrzeug fährt in die Waschanlage und wird gewaschen, wobei die Waschanlage für weitere Fahrzeuge gesperrt wird
Function Input	Fahrzeug
Control Input	-
Resource Input	Waschanlage Fahrer
Function Output	Fahrzeug gewaschen
Control Output	-
Resource Output	Waschanlage Arbeitszeit Fahrer Wartezeit
Duration	Durschnittl. Waschzeit 10 min
Ending Status	Waschanlage frei

Aktivitätstyp 'Bewegen'*Beispiel : Fahrzeug wegfahren*

Aktivität (Enterprise Activity)	
Type	Bewegen : Material oder Information
Identifizier	<i>EA3</i>
Name	Wegfahren
Description	Fahrzeug fährt nach dem Waschen weg und macht die Waschanlage frei
Function Input	Fahrzeug
Control Input	Fahrzeug fertig gewaschen
Resource Input	Fahrer
Function Output	Fahrzeug
Control Output	
Resource Output	Fahrer Wegfahrzeit
Duration Min / Max / Average	-
Ending Status	Prozess-Ende

Ereignis / Event

Ereignis (Event)	
Type	Material (Ankunft)
Identifizier	<i>EVI</i>
Name	Fahrzeug trifft ein
Description	Fahrzeug trifft an der Waschanlage ein
Triggers	Enterprise Activity 'Warten'
Resource	Fahrer
Object	Fahrzeug
Häufigkeit	Exponentialverteilung, Mean = 12

Ablaufregeln / Behavioural Rule Set

zur dynamischen Verknüpfung der Aktivitäten, nach Event/ES-Condition-Action Prinzip

WHEN (START WITH Event1)			
	<i>Fahrzeug trifft ein</i>	DO (EA1)	<i>Warten</i>
WHEN Ending Status <i>Waschanlage frei</i>			
AND <i>Fahrzeug verfügbar</i>		DO (EA2)	<i>Waschen</i>
WHEN Ending Status <i>Fahrzeug gewaschen</i>			
AND <i>Waschanlage frei</i>		DO (EA3)	<i>Wegfahren</i>

Literatur

- [CIMOSA-93] '*CIMOSA - Open System Architecture for CIM*'; ESPRIT Consortium AMICE, Springer-Verlag 1993, (ISBN3-540-56256-7), (ISBN 0-387-56256-7).
- [CIMOSA-96] '*CIMOSA - Open System Architecture for CIM*', *Technical Baseline*; Version 3.2, Veröffentlichung der CIMOSA Association e.V. (1996) Homepage : <http://www.cimosa.de>
- [CIMOSA-99] '*CIMOSA- Evaluation and Applications in Enterprise Engineering and Integration*', Herausg. K. Kosanke, F. Vernadat, M. Zelm, Computers in Industry, Vol. 40 Numbers 2-3 (1999)
- [Vernadat-96] F.B. Vernadat, *Enterprise Modelling and Integration, Principles and Applications*; Chapman and Hall, 1996, ISBN 0-412-60550-3
- [PACE-99] PACE 3.1, Benutzer-Handbuch, IBE Ingenieurbüro, Software and Simulation Engineering, D-85823 Glonn, Mai 1999, Homepage <http://www.ibepace.com>