

Visuelle-Sprachen-Projekt SoSe 2007

Handbuch RONEditor:

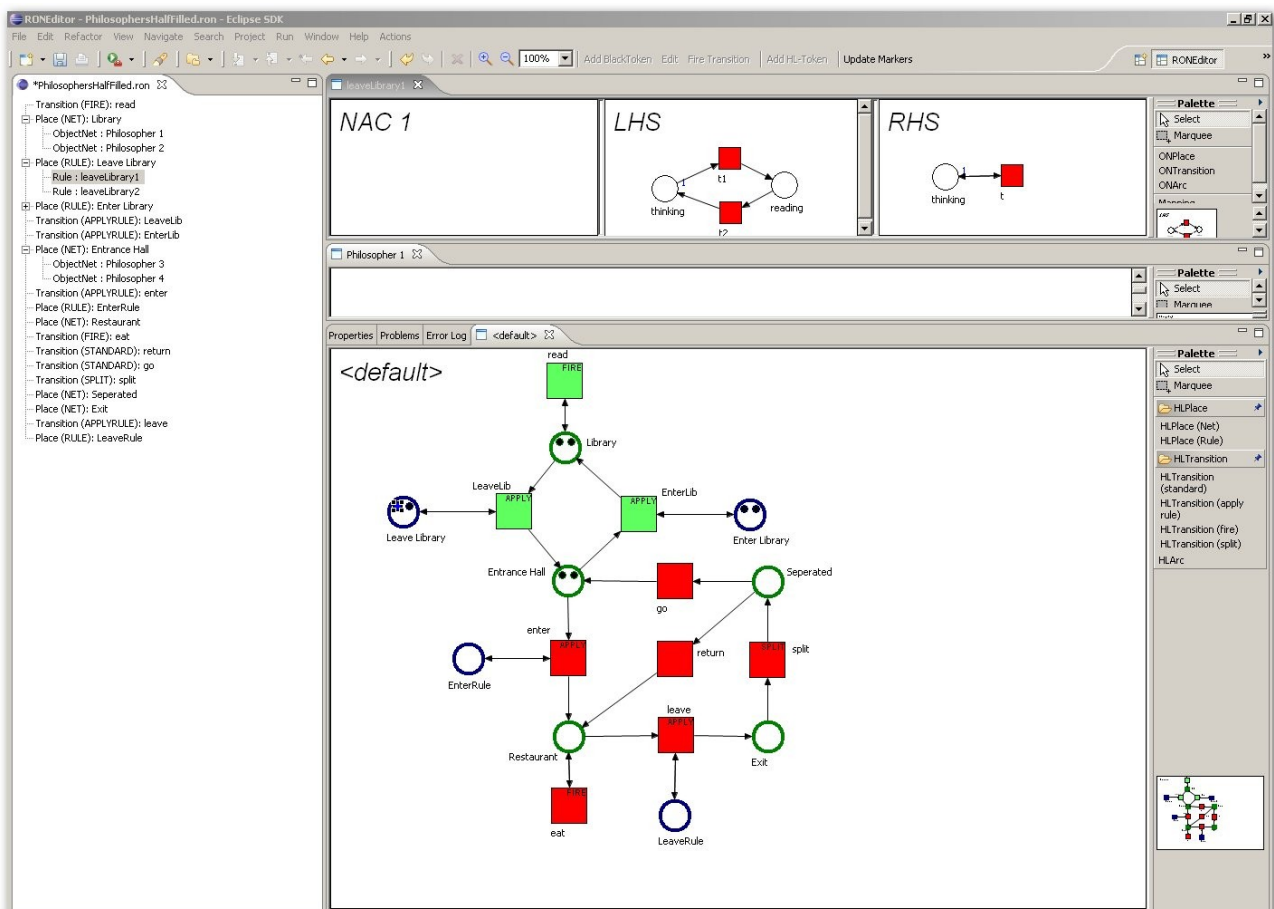


Abbildung1 -RON-Editor

Gruppe 1 bzw. 4:

Jochen Adamek
Conny Ullrich
Alexander Rein
Mario Mühlberg
Rabia Ghaber

Inhaltsverzeichnis:

- 1)Einführung-RON
- 2)RON-Editor
- 3)Objektnetz-Editor
- 4)Regel-Editor

1)Einführung RON:

Low-Level-Petrinetze(P/T-Netze) sind ein weitverbreiteter Formalismus zur Modellierung verteilter und nebenläufiger Systeme. P/T-Netze bestehen aus einer Menge von Transitionen, einer Menge von Stellen und gewichteten Kanten als Verbindung zwischen Stellen und Transitionen. Alle in eine Transition eingehenden Verbindungen sind der Vorbereich, alle von der Transition ausgehenden Verbindungen sind der Nachbereich. Eine Transition kann nur schalten, wenn alle Stellen im Vorbereich mindestens einen Token besitzen und der Wert grösser als das Kantengewicht ist.

Algebraische Definition von P/T-Netzen:

P/T-systems $PN = (P, T, pre, post, M^0)$

- ▶ a set of places P
- ▶ a set of transistions T
- ▶ weighted arcs $pre, post : T \rightarrow P^\oplus$
- ▶ the initial marking $M^0 \in P^\oplus$
- ▶ P/T-morphism $f = (f_P, f_T) : PN_1 \rightarrow PN_2$
with $f_P : P_1 \rightarrow P_2$ and $f_T : T_1 \rightarrow T_2$ s.t.

$$(1) \quad f_P^\oplus \circ pre_1 = pre_2 \circ f_T \text{ and } f_P^\oplus \circ post_1 = post_2 \circ f_T$$

$$(2) \quad f_P^\oplus(M_1^0|_p) \leq M_2^0|_{f_P(p)} \text{ for } p \in P_1$$

Abbildung2-Algebraische Definition von P/T-Netzen

Das Schaltverhalten von P/T-Netzen:

Transition t ist unter Markierung M aktiviert (enabled):

$$\triangleright \text{enabled}_A(PN, t) = \begin{cases} \text{true} & \text{if } t \in T, \text{pre}(t) \leq M \\ \text{false} & \text{else} \end{cases}$$

Transition t schaltet einmal (firing), was zu folgendem "neuen" Netz führt:

$$\triangleright \text{fire}_A(PN, t) = \begin{cases} (P, T, \text{pre}, \text{post}, M \ominus \text{pre}(t) \oplus \text{post}(t)) & \text{if } \text{enabled}_A(PN, t) = \text{true} \\ PN & \text{else} \end{cases}$$

Abbildung3-Schaltverhalten von P/T-Netzen

Die Abkürzung RON, eine vereinfachte Darstellungsform Algebraischer Higher-Order-Netze(AHONs), steht für ein „Rekonfigurierbares Objektnetz. Sogenannte „Tokens“ bilden Objektnetze oder DPO-Graphregeln.

Entscheidend ist, dass RONs keine formale Semantik durch eine Signatur und Algebra und keine Kanteninschriften besitzen. Die Typen der Transitionen sind dabei festgelegt. Entscheidend ist, dass ein EMF-Modell für rekonfigurierbare Objektnetze als Basis für den GEF-Editor gilt.

2)RON-Editor:

TreeView:

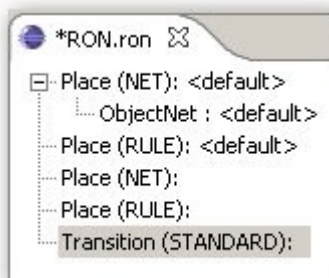


Abbildung4-TreeView

Der TreeView im linken Bereich des RON-Editors zeigt eine Übersicht über die existenten Stellen und Transitionen. In der Klammer ist angegeben, um welche Stelle(RULE oder NET) und Transitionen(STANDARD, APPLY_RULE, FIRE, SPLIT) es sich handelt. Dabei hat man die Möglichkeit mit einem Klick auf die rechte Maustaste einer Stelle, einen High-Level-Token(HLTToken) hinzuzufügen, die Stelle zu löschen, oder sie umzubenennen. Der im TreeView angezeigte Name taucht ebenfalls im RuleView oder im ObjectNetView auf.

Das Sternchen an der .ron-Datei signalisiert, dass eine Veränderung im Modell stattgefunden hat und dieses Modell bisher nicht gespeichert wurde. Das Speichern erfolgt entweder durch das Klicken auf das Diskettensymbol, oder man benutzt einen Short-cut(Standard-Tastenbelegung: strg+s).

Palette Ronview:

Die Palette befindet sich am rechten Rand der RON-Ansicht. Es erfolgt eine Unterteilung in „Select“, „Marquee“, „HLPlace“ und „HLTransition“. Mit einem Mausklick auf den Bereich „select“ wählen sie einen bestimmten Bereich im Ronview aus. Sie haben auch die Möglichkeit einen grösseren Bereich auszuwählen, indem sie auf eine Koordinate in der RonView-Ansicht klicken, die Maustaste gedrückt halten und den Bereich markieren. Der markierte Teil wird vor dem endgültigen Auswählen (linke Maustaste loslassen) durch ein gestricheltes Rechteck festgelegt.

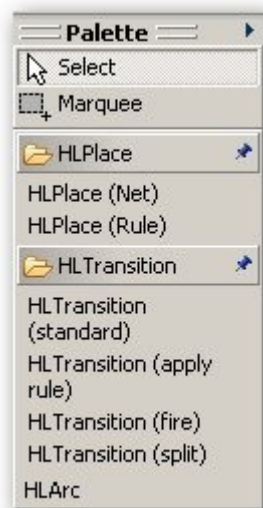


Abbildung5-Palette RonView

Allgemein ist es möglich durch einen Doppelklick der linken Maustaste eine entsprechende View auf dem ganzen RON-Editorbereich anzeigen zu lassen.

Möchte man wieder zur ursprünglichen Ansicht zurückkehren, kann man erneut mit einem Doppelklick auf den Namen der Ansicht (siehe Abbildung6) das Fenster verkleinern.



Abbildung6-Vergrößerung/Fenster

Erstellen von Objektnetzen als Stelle:

Wählen Sie in der Palette am rechten Rand des Eingabefeldes das Feld „HLPlace(Net)“ aus. Wenn sie die gewünschten Koordinaten der Stelle ausgewählt haben, erscheint ein grüner Kreis und unter dem Objekt eine Eingabemaske, in der sie den Namen der Stelle eingeben können.

Erstellen von Rules als Stelle:

Wählen Sie in der Palette am rechten Rand des Eingabefeldes das Feld „HLPlace(Rule)“ aus. Wenn sie die gewünschten Koordinaten der Stelle ausgewählt haben, erscheint ein blauer Kreis und unter dem Objekt eine Eingabemaske, in der sie den Namen der Stelle eingeben können.

Hinzufügen von HL-Tokens:

Wenn sie eine Stelle auswählen, haben sie die Möglichkeit, durch Betätigung des Feldes „Add HL-Token“ ein Token hinzuzufügen. Es erscheint ein kleines Menüfenster. Hier können sie den Namen des Tokens eingeben. Klicken sie auf OK, wenn sie mit der Benennung einverstanden sind und an der entsprechenden Stelle wird ein Token hinzugefügt. HLTOKEN-Namen dürfen nicht der leere String sein und keine „(, Klammern enthalten.

Wird eine bestimmte Zahl an Tokens auf einer Stelle erreicht, vergrößern sich die Kreise aller Stellen automatisch.

Durch einen Doppelklick auf einen Token öffnet sich eine neue View, in der sie wieder Stellen und Transitionen(entweder ein RuleView oder ein ObjectNetView) erstellen können.

Öffnet sich das Fenster, bleibt der Token markiert, sodass erkennbar ist, welche Ansicht gerade geöffnet ist.

Sie haben jedoch auch die Möglichkeit durch einen Doppelklick der linken Maustaste auf eine HL-Stelle, einen Token hinzuzufügen. Der Doppelklick initiiert das Öffnen eines Dialogfensters, indem sie den Namen des neuen Tokens und demnach einer neuen Regel oder eines Objektnetzes bestimmen können.

Sie können durch einen Doppelklick im TreeView den Namen des Tokens ändern. Dabei wird geprüft, ob hier eine Eindeutigkeit des Namens vorhanden ist. Sollte dies nicht der Fall sein, wird der alte Namen beibehalten.

Erstellen/Schalten einer Transition:

Es muss zwischen vier verschiedenen Transitionen unterschieden werden. Der Transitionstyp „STANDARD“ bewegt Objektnetze durch das HL-Netz und besteht aus den Funktionen „delete, copy, move, union“.

Übersicht HL-Transitionstyp STANDARD:

Vorbereich:

Höchstens eine Rule-Stelle, beliebig viele NET-Stellen

Nachbereich:

Beliebig viele RULE(NET)-Stellen, wenn mindestens eine RULE(NET)-Stelle im Vorbereich ist(sonst 0).

Aktivierung:

Mindestens ein Token auf jeder Vorbereichsstelle.

Schaltverhalten:

Von jeder Vorbereichsstelle wird ein Token entfernt. Auf alle NET-Stellen im Nachbereich wird die disjunkte Vereinigung aller Objektnetze, die dem Vorbereich entnommen wurden, gelegt.

Auf alle RULE-Stellen im Nachbereich wird die Regel, die dem Vorbereich entnommen wurde, gelegt.

Der Transitionstyp „FIRE“ schaltet eine Transition in einem Objektnetz.

Übersicht HL-Transitionstyp FIRE:

Vorbereich:

Eine NET-Stelle

Nachbereich:

Eine NET-Stelle(muss nicht unbedingt genau die gleiche NET-Stelle sein, die im Vorbereich ist)

Aktivierung:

Mindestens ein ON-Token liegt auf der Vorbereichsstelle, und mindestens eine Transition des Objektnetzes ist aktiviert.

Schaltverhalten:

Bei mehreren ON-Token: Auswahl eines Ons durch Benutzeraktion

Bei mehreren, aktivierten Transitionen: Auswahl durch Benutzeraktion

Beim ausgewählten ON wird die ausgewählte Transition geschaltet. Das resultierende ON wird auf alle NET-Stellen im Nachbereich gelegt.

Der Transitionstyp „APPLY_RULE“ wendet eine Netztransaktionsregel einmal auf ein Objektnetz an.

Übersicht HL-Transitionstyp APPLY_RULE:

Vorbereich:

Mindestens eine NET-Stelle, genau eine RULE-Stelle.

Nachbereich:

Beliebig viele NET-Stellen, beliebig viele RULE-Stellen.

Aktivierung:

Mindestens ein ON-Token liegt auf jeder Vorbereichs-NET-Stelle und mindestens ein Regeltoken liegt auf jeder Vorbereichs-RULE-Stelle.(und es existiert ein match von mindestens einer Regel in die Union aller ON-Token.)

Schaltverhalten:

Von jeder NET-Stelle im Vorbereich wird ein ON-Token entfernt.

Von jeder RULE-Stelle im Vorbereich wird ein Regel-Token entfernt.

Die Regel wird auf die Union aller entfernten ON-Token angewendet. Auf jede Nachbereichs-NET-Stelle wird das Resultat der Regelanwendung gelegt. Auf jede Nachbereichs-RULE wird die angewendete Regel gelegt.

Der Transitionstyp „SPLIT“ spaltet ein Objektnetz, das aus n Zusammenhangskomponenten besteht, auf in n entsprechende zusammenhängende Objektnetze.

Der HL-Transitionstyp „SPLIT“:

Vorbereich:

Genau eine NET-Stelle, keine RULE-Stelle.

Nachbereich:

Genau eine NET-Stelle, keine RULE-Stellen.

Aktivierung:

Mindestens ein ON-Token liegt auf der Vorbereichs-NET-Stelle.

Schaltverhalten:

Ein ON-Token der Vorbereichsstelle wird ausgewählt.

Besteht das ON aus mehreren Zusammenhangskomponenten, so wird auf der Nachbereichsstelle für jede einzelne Zusammenhangskomponente ein ON-Token erzeugt, das der Komponente entspricht.

Wie fügt man eigentlich eine Transition hinzu?:

Wenn sie eine Transition im RON-Editor hinzufügen wollen, klicken sie in der Palette im rechten Bereich auf eines der Felder der HLTransitionen. Haben sie die Koordinaten der Transition ausgesucht, bildet sich ein rotes Rechteck. Eine Transition vom Typ „STANDARD“ wird nicht namentlich gekennzeichnet, alle anderen Transitionstypen werden durch einen rechts oben im Rechtecksymbol befindlichen Namen klassifiziert.

Die rote Farbe signalisiert, dass die Transition derzeit nicht schalten kann. Ist das Schalten entsprechend der Bedingungen der verschiedenen Transitionstypen aktiviert, hat eine Transition eine grüne Farbe.

Das Feuern einer Transition erfolgt durch das Feld „Fire Transition“ oberhalb der Views. Markiert man eine grüne Transition, d.h. Eine Transition, die schalten kann, ist das Feld „Fire Transition“ freigegeben. Sie haben aber zusätzlich die Möglichkeit durch einen Doppelklick mit der linken Maustaste auf eine grüne und daher schaltfähige Transition, eine Transition schalten zu lassen.

Bei beiden Varianten öffnet sich ein Fenster mit den einzelnen Tokens, die beim Schalten angesprochen werden.

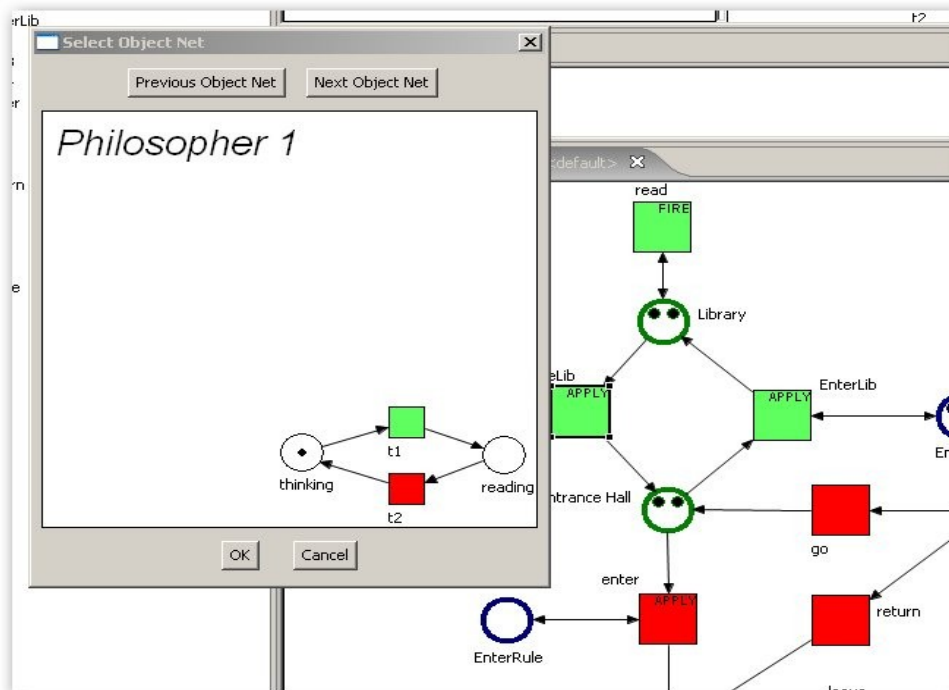


Abbildung7-ObjectNet-Fenster beim Schalten einer Transition

Abbildung7 zeigt ein solches geöffnetes Fenster eines Objektnetzes. Mit einem Klick der linken Maustaste auf das Feld „Next Object Net“ können sie einen weiteren Token auswählen und selbstverständlich durch Auswahl des Feldes „Previous Object Net“ zurückkehren. Im RonView bleibt die gerade angesprochene Transition markiert. In der Abbildung7 sehen sie, wie die grüne Transition „LeaveLib“ markiert ist.

„Philosopher 1 „ steht für den Namen des HLTokens. Bestätigen müssen sie mit „OK“, abbrechen können sie jederzeit durch Klicken des Feldes „Cancel“ oder des x-Icons rechts oben in der Ecke des Fensters..

Möchten sie das Fenster individuell platzieren, um vielleicht eine bessere Sicht auf den TreeView oder das HLNetz zu haben, können sie das Fenster beliebig verschieben.

Ein geöffnetes Fenster bezüglich einer Regel sieht wie folgt aus:

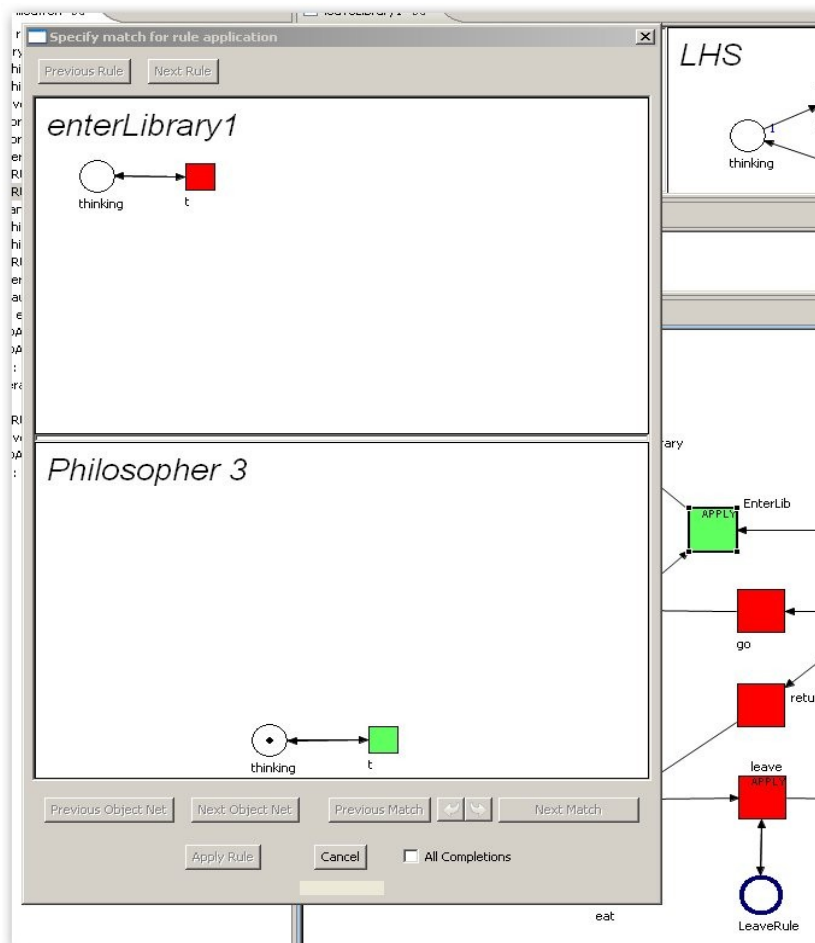


Abbildung8-Rule-Fenster beim Schalten einer Transition

Sie sehen auch an der Abbildung8 durch eine Markierung der entsprechenden Transition, welche Transition gerade angesprochen wird. Das Fenster verlassen können sie durch das Feld „Cancel“. Ebenfalls können sie zwischen den einzelnen Regeln hin und her wechseln. „Calculate Completions“ bezieht sich auf das Finden eines Matches durch AGG.

Verbinden von Stellen und Transitionen:

Durch das Feld „HLArc“ in der Palette kann man Stellen und Funktionen miteinander verbinden. Verbindungen zwischen zwei Transitionen oder zwischen zwei Stellen sind nicht erlaubt.

Verschieben von Tokens:

Sie haben die Möglichkeit, HL-Tokens von einer Stelle in eine andere Stelle zu verschieben, indem sie ein Token mit der linken Maustaste anklicken, gedrückt halten und in die entsprechende Stelle verschieben, wobei sie darauf achten müssen, dass ein Token als RULE nur in eine RULE-Stelle verschoben werden kann. Das Gleiche gilt ebenfalls für Tokens als Objektnetze.

3)Objektnetz-Editor:

Palette:

Die Palette ist das Grundgerüst des Objektnetz-Editors und befindet sich rechts von der Zeichnungsfläche.

(siehe nachfolgender Screenshot)



Abbildung9-Palette ObjectNetView

Erstellen von Places:

Für die Erstellung eines Places steht in der Palette am rechten Rand das Feld „ONPlace“ zur Verfügung. Klicken Sie dieses Feld an und platzieren Sie entsprechend Ihrer Wünsche den Knoten im Feld. Durch die Realisierung von DirectEdits(siehe Abschnitt DirectEdit) wird nach Festlegung des Ortes einer Stelle, eine Maske geöffnet, in der der Name der Stelle eingegeben werden kann. Der Default-Wert der Token wird auf 0 gesetzt.

Erstellen von Transitionen:

Für die Erstellung einer Transition(„ONTransition“) steht in der Palette das Feld „ONTransition“ zur Verfügung. Klicken Sie auf dieses Feld und sie können anschliessend auf dem Feld eine Transition erstellen. Haben Sie sich für einen Ort der Transition festgelegt, erscheint eine Eingabemaske, in der der Name der Transition eingegeben werden kann..

Zur besseren Differenzierung zwischen Transitionen, die schalten können und nicht schalten können, wird eine Transition farblich markiert. Eine grüne Transition lässt ein Schalten zu, eine rote Farbe der Transition unterbindet das Schalten an dieser Transition. Der Pfeil deutet auf die Schaltungsrichtung hin.

Verbinden von Transitionen und Places:

Für diese Funktion steht Ihnen in der Palette das Feld „ONArc“ zurVerfügung. Klicken Sie dieses Feld an, erscheint ein Dialogfenster und die Eingabe eines Kantengewichtes ist erforderlich. Der Default-Wert liegt bei 1, wenn der Benutzer keinen Wert, 0 oder irgendwelche anderen Symbole in die Eingabemaske eingibt.

Eine Verbindung zwischen zwei Transitionen und zwischen zwei Stellen ist nicht erlaubt. Der Mauszeiger gibt durch ein Symbol an, welche Verbindungen erlaubt sind und welche nicht.

Alle konstruierten Symbole sind des weiteren in einer Miniaturansicht am rechten, unteren Rand einsehbar.

Hinzufügen von Token:

Bis zu einer Zahl von 3 werden Token als kleine punkartige Symbole in den Stellen angezeigt. Ab einer Anzahl von 4 wird der Zahlenwert angezeigt. Indem man mit der rechten Maustaste auf eine Stelle klickt, hat man die Möglichkeit über die Funktion „Add Black Token“ beliebig viele Token hinzuzufügen. Eine zweite Alternative ist die Funktion „Add Black Token“ in der Toolbarleiste oberhalb der graphischen Umsetzung. Noch benienungsfreundlicher ist das Hinzufügen von einem Token durch einen Doppelklick auf die entsprechende Stelle. Ab einer Anzahl von 4 Token, wird ein Zahlenwert angegeben, was durch folgende Abbildung illustriert wird:



Abbildung10-Token in ObjectNetView

Es erfolgt nicht wie im Ronview eine Vergrößerung der Stellen.

Löschen/Update:

Um eine Transition, eine Verbindung, eine Stelle oder mehrere Bereiche zu löschen, muss man auf das Feld „Select“ in der Palette klicken. Möchte man den Namen oder Kantengewichte und Tokenanzahlen verändern, hat man die Möglichkeit mit der rechten Maustaste einen Properties Dialog auszuwählen. Sie können die Parameter der einzelnen Symbole ebenfalls verändern, indem sie die Funktion „Edit“ in der Toolbarleiste oberhalb der graphischen Realisierung auswählen, nachdem sie zuvor das entsprechende Symbol mit der linken Maustaste angeklickt bzw. aktiviert haben.

Schalten von Transitionen:

Das Schalten von Transitionen erfolgt ebenfalls über die linke Maustaste und anschliessendes Auswählen der Funktion „Fire ONTRransition“ oder über das Anklicken der Funktion „Fire“ in der Toolbarleiste oberhalb der graphischen Umsetzung. Dabei können Transitionen nur schalten, bzw. diese Funktion ist nur aktiviert, solange die Transitionen grün farblich gekennzeichnet sind. Ausschlaggebend sind die Kriterien für Petrinetze.

Features:

BendPoints:

Mit diesem Feature ist es möglich, Verbindungen zwischen Transitionen und Stellen durch Festlegung neuer Punkte individuell zu gestalten. Durch die Funktion „Select“ in der Palette hat man die Möglichkeit nach der Konstruktion einer Verbindung(ONArc), jene Verbindung auszuwählen. In der Mitte der Verbindung existiert ein sogenannter BendPoint, denn man neu setzen kann. Diesen Vorgang kann man beliebig wiederholen, denn jedes neue Setzen der Punkte impliziert zwei neue BendPoints. Sie haben aber auch die Möglichkeit durch einen Rechtsklick auf einer Verbindung(Arc) durch Bestätigen des Feldes „Delete BendPoints“ alle BendPoint-Punkte zu

löschen.

DirectEdit:

Um Fehler in der Benennung der Transitionen und Stellen benutzerfreundlich zu editieren, hat man die Möglichkeit, durch einen einfachen Klick mit der linken Maustaste, die Eingabemaske erneut aufzurufen.

Verschiebbare Labels:

Sie haben die Möglichkeit, durch Anklicken der Namen von Transitionen und Stellen, den Text beliebig zu verschieben. Sie müssen dafür den Text markieren, die linke Maustaste gedrückt halten und können den Text nun neu platzieren.

AutoLayout:

Durch das Klicken der rechten Maustaste im RuleView und im ObjectNetView hat man die Möglichkeit, alle Symbole automatisch zu strukturieren, indem man das Feld im Kontextmenu „AutoLayout“ auswählt.

4)Regel-Editor:

Zu Beginn stellt sich die Frage, was Netztransformationen überhaupt sind. Es sind Bestimmungen, mit denen ein Modell durch die Anwendung von bestimmten Regeln verändert werden kann. Die folgende Abbildung verdeutlicht die Strukturen einer Netztransformation:

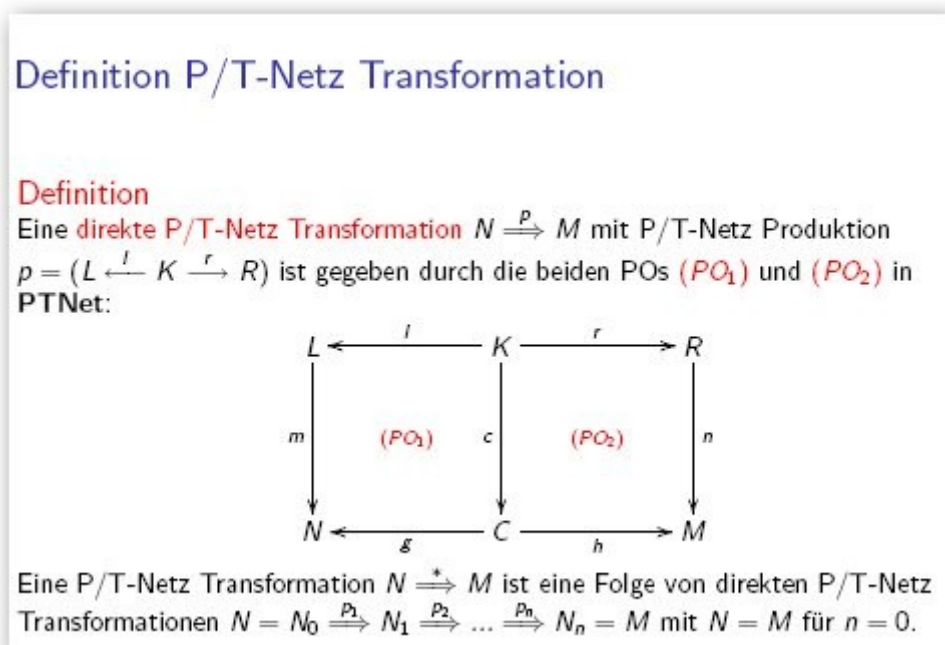


Abbildung11-Definition P/T-Netz-Transformation

ON-Morphismen(Familie von Abbildungen) sind wiefolgt aufgebaut:

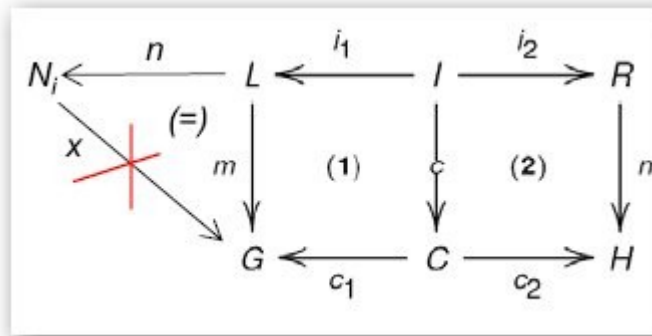


Abbildung12-ON-Morphismen

Der Morphismus von L nach N ist formal total, wir beschränken uns aber auf einen partiellen Morphismus. Partielle Morphismen können links injektiv und rechts allgemein dargestellt werden. Das Interface I ist ein Untergraph von L. Dadurch werden Kanten immer mit ihren Endknoten abgebildet. Die Attribute müssen nicht vollständig gesetzt sein.

Im ersten Schritt erfolgt ein Löschen des Elementes, im zweiten Schritt erfolgt das Hinzufügen von Elementen.

Das Anwenden der Regeln erfolgt durch Attributierte Graph-Grammatiken, kurz AGG genannt. AGG. Wenn sie sich intensiver mit AGG beschäftigen wollen, dann können wir Ihnen den Link am Ende dieses Handbuches zu AGG empfehlen.

Palette RuleView:

Für eine Regelansicht existiert eine eigene Palette. Sie unterscheidet sich von der Palette in der Objektanzicht nur durch das Feld „Mapping“, welches für eine Verbindung(Mapping) einer Stelle oder Transition einer Regel in eine andere Regel zuständig ist. Näheres dazu ist im folgenden Kapitel „Mapping“ nachzulesen.

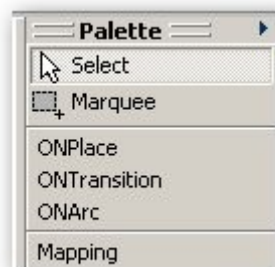


Abbildung13-Palette RuleView

Mapping:

Für das Mapping stehen 3 Regelbereiche zur Verfügung („NAC, LHS, RHS“) LHS steht dabei für die linke Regelseite, RHS für die rechte Regelseite und NAC für negative Anwendungen. Dabei sind nur Abbildung(Bildung von Morphismen) von LHS nach RHS, von LHS nach NAC und von RHS nach NAC möglich.

Dadurch ist es dem Benutzer gestattet, Teile wiederzuverwenden, zumal in einer NAC und der rechten Regelseite T viele Teile der linken Regelseite L wieder auftreten.



Abbildung 14-Regelansichten

Eine Abbildung zwischen den verschiedenen Regeln muss typverträglich sein. Es dürfen lediglich Stellen auf Stellen und Transitionen auf Transitionen abgebildet werden. Ausserdem müssen die Vor- und Nachbedingungen der Transitionen überprüft werden.

Im Übrigen erfolgt lediglich ein injektives Mapping. Dabei werden zugehörige Stellen bzw. Transitionen durch einen gleichen Zahlenwert gekennzeichnet. Um das Mapping zu vereinfachen, existiert die Möglichkeit, ganze Petrinetze von einer Regelseite auf die andere Regelseite zu kopieren. Das Auswählen eines Petrinetzes oder Teilnetzes und anschliessendes Klicken der rechten Maustaste ermöglicht die Auswahl der „Copy“-Befehle.

Negative Regelanwendungen können durch die Auswahl „Create NAC“ im Kontextmenu erstellt und durch die Funktion „Delete NAC“ wieder gelöscht werden.

Das Mapping einer Stelle auf eine andere Stelle bzw. analog dazu beim Mapping von Transitionen erfolgt, indem man in der Palette auf die Funktion „Mapping“ klickt, die entsprechende Ausgangsstelle durch ein Klicken mit der linken Maustaste und entsprechend die Eingangsstelle auswählt. Die Kennzeichnung der Nummern in blauer Schriftfarbe steht für die zugeordneten Stellen bzw. Transitionen. Diese Nummerierungen werden „Marker“ genannt. Bei der Umbenennung einer Regel ist gegebenenfalls darauf zu achten, dass sich gegebenenfalls die Marker ändern. Beim Löschen von HLNodes oder ONNodes(Transitionen) werden alle Marker von nicht mehr existierenden Regeln gelöscht. Ein Konsistenzcheck aller Rules ist notwendig, die zu diesem Zeitpunkt keine Marker mehr besitzen. Das Hinzufügen bzw. Löschen von NACs erfordert das Entfernen aller Marker der Regel und einen Konsistenzcheck der kompletten Regel. Treten Probleme beim „Mapping“ auf, werden Problemmeldungen im Eclipse-Problem-View angezeigt.

Property View:

Dieses Fenster ist unterhalb der graphischen Darstellung einsehbar. Angegeben werden die Parameter der einzelnen, konstruierten Symbole(ONNode, ONTransition, ONArc) und die dazugehörigen x/y-Koordinaten, sowie der Name. Wichtig ist, dass man eine Stelle oder eine Transition mit der linken Maustaste anklickt und dadurch aktiviert.

Im ObjectNetView wird u.a. auch die Anzahl der Token angegeben. Verschiebt man eine Transition oder eine Stelle, ändern sich natürlich auch die x-bzw. Y-Koordinaten.

Property	Value
x	0
y	0
name	newObjectNet6
labelx	0
labely	30

Abbildung15-PropertyView

Testfallbeispiel:

Wir möchten Ihnen kurz ein gängiges Beispiel zeigen, welches verschiedene Abläufe und die Funktionsweise der einzelnen Transitionstypen kontrolliert.

Es ist eine Erweiterung der Speisenden Philosophen um eine Bibliothek, in der die Philosophen zusätzlich zum Essen und Denken, Lesen können. In der Stelle „Entrance Hall“ werden beide Netze zusammengefasst. Es wird durch die farbliche Unterscheidung der Transitionen gut ersichtlich, welche Transitionen schalten können und welche nicht.

Es bieten sich weitere Beispiele an. Ein weiteres Testbeispiel ist die Darstellung „Verteilter Konsumenten/Produzenten“.

Dabei sind Produzenten und Konsumenten nicht verbunden. Für die Abwicklung des Verkaufs treffen sie sich und verbinden ihre Netze. Anschliessend trennen sie sich wieder.

Desweiteren könnte man ein sogenanntes Katastrophen-Szenario erstellen.

Mehrere Team-Mitglieder, die mit mobilen Geräten kommunizieren können, arbeiten in einem Katastrophengebiet. Ein Team-Leader koordiniert die Aufgaben des Teams.

Wichtig ist noch zu erwähnen, dass die Aufgaben einzeln an die Team-Mitglieder verteilt werden. Sie werden jedoch in einem gemeinsamen Workflow abgearbeitet.

Haus der Philosophen:

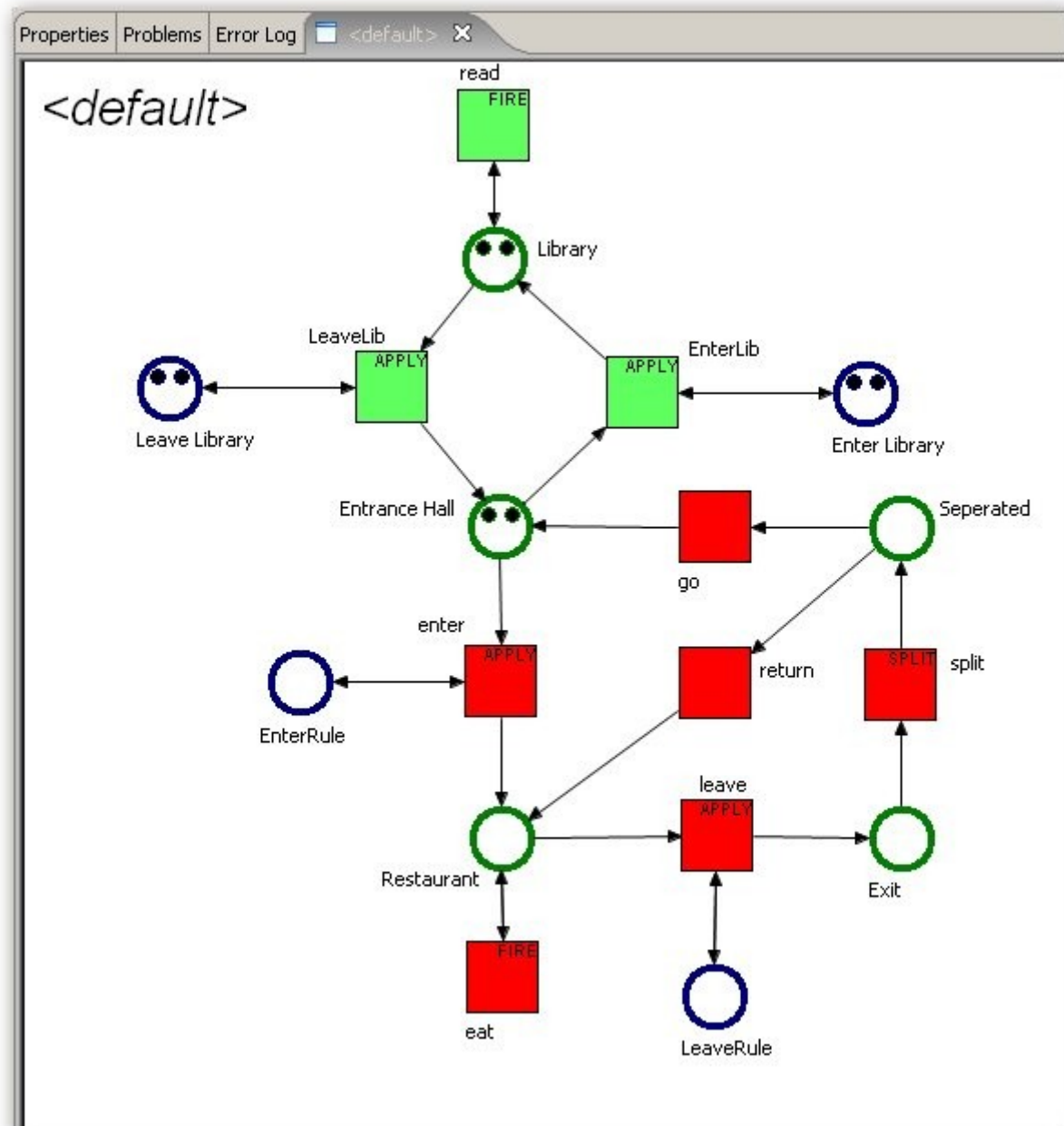


Abbildung16-Testbeispiel

Empfehlenswerte Links:

<http://tfs.cs.tu-berlin.de/vila/> (Homepage des Projektes „Visuelle Sprachen“ des Fachbereiches Theoretische Informatik / Formale Spezifikation (TFS))

<http://www.eclipse.org>

<http://eclipse-plugins.2y.net/eclipse/plugins.jsp>

<http://www.eclipse.org/emf>

<http://www.eclipse.org/gef>

<http://tfs.cs.tu-berlin.de/vila/literatur.html>
(weitere Plug-Ins(z.B.Dialoge: VisualEditor, Jigloo))

<http://tfs.cs.tu-berlin.de/agg/> (Attributierte Graph-Grammatiken)

Vielen Dank!