

Prozess-Simulation in KMU

Entwicklung und Analyse eines Petrinetz-basierten Baukastensystems
als Instrument zur Modellierung und Simulation
von Produktions- und Logistikprozessen –
speziell ausgelegt zur Unterstützung
von kleinen und mittleren Unternehmen (KMU)

Anhang zur

Dissertation

zur Erlangung des akademischen Grades eines
Doktors der Wirtschaftswissenschaften (Dr. rer. pol.)
durch die Fakultät für Wirtschaftswissenschaften der
Universität Duisburg-Essen, Campus Essen

vorgelegt von

Diplom-Wirtschaftsingenieurin (FH), Master of Systems Engineering

Stefanie Rudel aus Unterhaching

Essen, 2016

Anhang A1

Leitfaden für Experteninterview

Leitfaden für Experteninterview

Vorlage

Autor: Steffi Rudel

Version: 1.0

Datum: 20.03.2014

Leitfadeninterview

Rahmendaten

Unternehmen	
Anzahl Mitarbeiter im Unternehmen	
Standort Unternehmen	
Herstellung von Metallerzeugnissen	Ja Nein
Ort, Datum Interview	
Name Interviewpartner 1	
Funktion Interviewpartner 1	
Name Interviewpartner 2	
Funktion Interviewpartner 2	
Name Interviewpartner 3	
Funktion Interviewpartner 3	

Frage 2	Offen
Wie läuft denn ein typischer Auftrag durch Ihre Produktion – vom Auftragsingang bis zur Auslieferung?	
Antwort	
Mögliche Konkretisierungsfragen <ul style="list-style-type: none"> • Welche Stationen durchläuft der Auftrag denn? • Werden die unfertigen und fertigen Erzeugnisse irgendwo zwischengelagert, z.B. wenn eine Maschine gerade belegt ist? 	
Antwort Konkretisierung	
Ziel: Offene Einstiegsfrage. Generellen Ablauf kennenlernen.	

Frage 4	Offen
Wo sehen Sie das größte Optimierungspotential Ihrer Produktion und Logistik?	
Antwort	
<p>Mögliche Konkretisierungsfragen</p> <ul style="list-style-type: none"> • Wo könnte man am ehesten etwas verbessern? • Welcher Bereich ist momentan am „schlechtesten“? 	
Antwort Konkretisierung	
Ziel: Einschätzung eines Experten bekommen, wo Potential zur Optimierung gesehen wird.	

Frage 5	Offen
Was sind die größten Probleme in Ihrer Produktion?	
Antwort	
Mögliche Konkretisierungsfragen <ul style="list-style-type: none"> • Wo gibt es immer wieder Probleme? • Wo hängt es immer wieder? • Was stört Sie besonders? 	
Antwort Konkretisierung	
Ziel: Probleme in der Produktion herausfinden.	

Frage 6	Offen
Was sind die größten Probleme in Ihrer Logistik?	
Antwort	
Mögliche Konkretisierungsfragen <ul style="list-style-type: none"> • Wo gibt es immer wieder Probleme? • Wo hängt es immer wieder? • Was stört Sie besonders? 	
Antwort Konkretisierung	
Ziel: Probleme in der Produktionslogistik herausfinden.	
Anmerkung: Selbst aussortieren, was zur Produktionslogistik gehört und was nicht. Interviewpartner sollte das nicht müssen.	

Frage 7	Offen
Was macht bei Produktionsaufträgen die meisten Schwierigkeiten?	
Antwort	
Mögliche Konkretisierungsfragen <ul style="list-style-type: none"> • Wo gibt es immer wieder Probleme? • Wo muss man immer wieder korrigieren? 	
Antwort Konkretisierung	
Ziel: Schwierigkeiten in der Produktion aufdecken.	
Anmerkung: Ziemlich ähnlich wie Frage 4, jedoch speziell auf Auftrag bezogen und daher etwas anders formuliert.	

Frage 8	Offen
Erinnern Sie sich an einen Produktionsauftrag, der „total in die Hose ging“ – was war da passiert? Woran lag das?	
Antwort	
Mögliche Konkretisierungsfragen <ul style="list-style-type: none"> • Vielleicht gab es einen Auftrag, der ein ziemliches Verlustgeschäft war? • Oder einen wo der Termin nicht gehalten werden konnte? • Was war da passiert? • Woran lag das? 	
Antwort Konkretisierung	
Ziel: Negativbeispiel zur Identifizierung der Probleme nutzen.	

Frage 9	Offen
Erinnern Sie sich an einen Produktionsauftrag, der besonders gut lief – was war da passiert? Woran lag das?	
Antwort	
Mögliche Konkretisierungsfragen <ul style="list-style-type: none">• Vielleicht gab es einen Auftrag, der ein ziemlich gutes Geschäft war?• Oder der Kunde besonders zufrieden war?• Was war da passiert?• Woran lag das?	
Antwort Konkretisierung	
Ziel: Positivbeispiel zur Identifizierung von Problemen nutzen.	

Frage 10		Bewertung, Offen				
Wie wichtig sind die folgenden Aspekte für Sie, bewerten von 1 (unwichtig) bis 5 (sehr wichtig)?						
A	Niedrige Durchlaufzeiten der Aufträge	1	2	3	4	5
B	Hohe Lieferfähigkeit (inwieweit es dem Vertrieb möglich ist, vom Kunden gewünschte Liefertermine unter Beachtung der Produktionsgegebenheiten zusagen zu können)	1	2	3	4	5
C	Hohe Liefertreue (in welchem Maße die bei der Auftragserteilung zugesagten Termine realisiert werden können)	1	2	3	4	5
D	Hohe Auslastung der Mitarbeiter	1	2	3	4	5
E	Gleichmäßige Auslastung Mitarbeiter (nicht stark schwankend)	1	2	3	4	5
F	Hohe Auslastung der Maschinen	1	2	3	4	5
G	Gleichmäßige Auslastung Maschinen (nicht stark schwankend)	1	2	3	4	5
H	Lagerbestände auf ein möglichst niedriges Niveau zu reduzieren	1	2	3	4	5
I	Umlaufbestände auf ein möglichst niedriges Niveau zu reduzieren (z.B. unfertige Erzeugnisse)	1	2	3	4	5
K	Bestand Rohmaterial optimieren (reduzieren aber trotzdem zum richtigen Zeitpunkt verfügbar haben)	1	2	3	4	5
L	Bestand Hilfsmaterial optimieren (reduzieren aber trotzdem zum richtigen Zeitpunkt verfügbar haben)	1	2	3	4	5
M	Bestand fertige Erzeugnisse optimieren (reduzieren aber trotzdem zum richtigen Zeitpunkt verfügbar haben)	1	2	3	4	5
N	Material-Warteschlangen reduzieren	1	2	3	4	5
O	Flaschenhalse im Produktionsprozess identifizieren	1	2	3	4	5
P	Auftragsreihenfolgen planen, um effizienter abzuwickeln	1	2	3	4	5
R	Puffergrößen optimieren (Lager zwischen den einzelnen Maschinen)	1	2	3	4	5
S	Besser vorher wissen, wann welche Maschine belegt ist	1	2	3	4	5
T	Besser vorher wissen, wann welcher Mitarbeiter belegt ist	1	2	3	4	5
U	Produktionsprozess anschaulich abbilden	1	2	3	4	5
W	Was noch?	1	2	3	4	5

X	Was noch?	1	2	3	4 5

Antworten Hinterfragungen

Ziel: Gewichtung der einzelnen Zielgrößen hinterfragen.

Anmerkung: Hilfstafel hinlegen. Markante Antworten jeweils hinterfragen, warum das so gesehen wird.

Frage 11		Bewertung, Offen
Kennen/messen Sie folgende Zielgrößen in Ihrem Unternehmen, wenn ja wie oft durchschnittlich (täglich, wöchentlich, monatlich, jährlich, nie)?		
A	Ausbringmenge (herausfinden ob schwankt)	Tag Woche Monat Jahr nie
B	Bestand Rohmaterial	Tag Woche Monat Jahr nie
C	Bestand Hilfsmaterial (was ist das z.B.?)	Tag Woche Monat Jahr nie
D	Bestand fertige Erzeugnisse	Tag Woche Monat Jahr nie
E	Durchlaufzeit der Aufträge	Tag Woche Monat Jahr nie
F	Auslastungsgrad Maschinen	Tag Woche Monat Jahr nie
G	Auslastungsgrad Mitarbeiter	Tag Woche Monat Jahr nie
H	Termintreue bei Produktionsaufträgen	Tag Woche Monat Jahr nie
I	Lagerverweilzeit fertige Erzeugnisse (wie lange bleiben fertige Erzeugnisse im Lager)	Tag Woche Monat Jahr nie
K	Was noch?	Tag Woche Monat Jahr nie
L	Was noch?	Tag Woche Monat Jahr nie
Antworten Hinterfragungen		
Ziel: Feststellen, welche Kennzahlen vorhanden sind.		
Anmerkung: Hilfstafel hinlegen. Antworten ggf. hinterfragen, z.B. wie das gemessen wird oder warum nicht gemessen wird. Im nächsten Schritt anschauen.		

Frage 12	Offen
Welches sind die wichtigsten Maschinen in Ihrem Unternehmen?	
Antwort	
Mögliche Konkretisierungsfragen <ul style="list-style-type: none"> • Warum sind die für Sie wichtig? Teuer in der Anschaffung? Teuer im Unterhalt? Teuer pro produzierte Einheit? • Wenn welche Maschine defekt wäre, wo wäre das das größte Problem? 	
Antwort Konkretisierung	
Ziel: Flaschenhalse herausfinden.	

Frage 13	Offen
Welche Maschinen hat ein Unternehmen Ihrer Branche und Größe typischerweise?	
Antwort	
Mögliche Konkretisierungsfragen <ul style="list-style-type: none"> • Welche braucht man denn unbedingt? • Ohne welche geht es denn gar nicht? 	
Antwort Konkretisierung	
Ziel: Herausfinden welche Maschinen modelliert werden müssen.	

Bereich Prozess-Simulation

Frage 14	Offen
Was haben Sie bisher von Prozess-Simulation gehört?	
Antwort	
Mögliche Konkretisierungsfragen <ul style="list-style-type: none">• Was stellen Sie sich darunter vor?• Wo/Wie haben Sie davon gehört?	
Antwort Konkretisierung	
Ziel: Hintergrund dieses Unternehmens bezüglich Prozess-Simulation erfragen.	

Frage 15	Offen
Haben Sie in Ihrem Unternehmen schon mal etwas mit Prozess-Simulation gemacht?	
Antwort	
Mögliche Konkretisierungsfragen <ul style="list-style-type: none">• Haben Sie schon mal etwas davon gehört?• Warum besteht da kein Interesse bei Ihnen?	
Antwort Konkretisierung	
Ziel: Hintergrund dieses Unternehmens bezüglich Prozess-Simulation erfragen.	

Frage 16	Offen
Was stellen Sie sich vor, was die Prozess-Simulation leisten kann?	
Antwort	
Mögliche Konkretisierungsfragen <ul style="list-style-type: none"> • Welche Vorteile denken Sie bringt es? • Und warum? 	
Antwort Konkretisierung	
Ziel: Einschätzung bekommen, wie Interviewpartner den Nutzen von Prozess-Simulation einschätzt.	

Hilfstafel zur Bewertung der Frage 10

1	2	3	4	5	
unwichtig				Sehr wichtig	

Tag Woche Monat Jahr nie

Anhang A2

Experteninterview Heiland

Leitfadeninterview Nachträglich erfasstes ist *kursiv* dargestellt

Rahmendaten

Unternehmen	Max und Franz Heiland GmbH
Anzahl Mitarbeiter im Unternehmen	Ca. 20 Festangestellte, dazu ca. 10 Aushilfen (auf 400,- EUR-Basis)
Standort Unternehmen	Geschwister-Scholl-Straße 29, 82008 Unterhaching
Herstellung von Metallerzeugnissen	<u>Ja</u> Nein
Ort, Datum Interview	Unterhaching, 27.3.2014
Name Interviewpartner 1	Max Heiland (Senior)
Funktion Interviewpartner 1	Geschäftsführer
Name Interviewpartner 2	Max Heiland (Junior)
Funktion Interviewpartner 2	Angestellter, zukünftiger Geschäftsführer

Bereich Produktion & Logistik

	Frage 1	Offen
	Bitte erzählen Sie mir erst einmal was für Metallerzeugnisse Ihr Unternehmen herstellt.	
	Antwort	
H1.1	- Drehteile, Frästeile aus Alu, Stahl, Edelstahl, Kunststoff, Messing	
H1.2	- Im Auftrag, nach Zeichnung	
H1.3	- Augenlaser, Medizin, Kamerastative	
H1.4	- <i>Stückzahl 1 -... (Preise sind gestaffelt)</i>	
	Ziel: Offene Einstiegsfrage. Generelle Erzeugnisse kennenlernen.	

	Frage 2	Offen
	Wie läuft denn ein typischer Auftrag durch Ihre Produktion – vom Auftragseingang bis zur Auslieferung?	
	Antwort	
H2.1	- Anfrage mit Zeichnung	
H2.2	- Bestellung schriftlich	
H2.3	- Planung mit Access	
H2.4	- Auftragsbestätigung mit Terminzusage (Lieferzeit wird von Access berechnet,	
H2.5	wird aber nicht genutzt, entscheidet Geschäftsführer „aus dem Bauch“)	
H2.6	- Materialbestellung	
H2.7	- Maschinenbelegungsliste wird gedruckt und an Mitarbeiter gegeben (ca. alle 14	
H2.8	Tage), dieser entscheidet was wo wann produziert wird braucht aber ok vom	
H2.9	Geschäftsführer	
H2.10	- Bearbeiten, wenn andere Stationen benötigt evtl. „Zwischenlager“ (kein Platz)	
H2.11	- Oberflächenbearbeitung (wird extern gemacht, wird teilweise geholt, teilweise	
H2.12	hingebacht, je nach Art der Behandlung gibt es einen oder mehrere Partner-	
H2.13	Unternehmen zur Auswahl)	
H2.14	- Evtl. nochmal anschauen	
H2.15	- Evtl. nachbearbeiten oder montieren	
H2.16	- Liefern (per Spedition oder per Post)	
	Konkretisierung	
H2.17	Bei Neuteil muss <i>CNC</i> programmiert werden	
	Ziel: Offene Einstiegsfrage. Generellen Ablauf kennenlernen.	

	Frage 3	Geschlossen
	Produzieren Sie nur „auf Auftrag“?	
H3.1 H3.2 H3.3	Antwort - Bei kleinen Mengen unter Umständen mehr, auf Vorrat, weil „die wieder kommen“ (nach ca. zehn Jahren wird schon mal was weggeworfen) - „Stange fertig“	
H3.4 H3.5 H3.6	Konkretisierung - Weil man dann auf Engpass <i>vom Kunden</i> reagieren kann und Rüstkosten spart - Teile alle speziell Kunden-/Auftragsbezogen - <i>Kein eigenes „Produkt“</i>	
	Ziel: Herausfinden ob über die Produktion selbst bestimmt werden kann oder ausschließlich vom Kunden getrieben wird.	

	Frage 4	Offen
	Wo sehen Sie das größte Optimierungspotential Ihrer Produktion und Logistik?	
H4.1	Antwort Platz, Lagerhaltung (Platz), <i>zugesagte Termine halten können</i>	
	Ziel: Einschätzung eines Experten bekommen, wo Potential zur Optimierung gesehen wird.	

	Frage 5	Offen
	Was sind die größten Probleme in Ihrer Produktion?	
H5.1	Antwort Platzproblem (treten sich auf die Füße)	
	Konkretisierungsfrage Wo hängt es immer wieder?	
H5.2 H5.3	Antwort Konkretisierung Eine Maschine ist überlastet, mit Platz könnte eine zweite <i>dieses Typs</i> angeschafft werden	
	Ziel: Probleme in der Produktion herausfinden.	

	Frage 6	Offen
	Was sind die größten Probleme in Ihrer Logistik?	
H6.1 H6.2 H6.3	Antwort - Maschinen sind teilweise im Keller, muss viel getragen werden - Dass keine „Fließfertigung“ ist, <i>Material in vorderste Maschine rein und an hinterster Maschine Produkt raus, dass Maschinen richtig stehen</i>	
	Ziel: Probleme in der Produktionslogistik herausfinden.	
	Anmerkung: Selbst aussortieren, was zur Produktionslogistik gehört und was nicht. Interviewpartner sollte das nicht müssen.	

	Frage 7	Offen
	Was macht bei Produktionsaufträgen die meisten Schwierigkeiten?	
H7.1 H7.2	Antwort Wenn viele Aufträge <i>im Eingangskorb</i> „warten“ auf Systemeingabe, „schlechtes Gefühl“ <i>wenn dann noch ein Auftrag kommt</i>	
	Ziel: Schwierigkeiten in der Produktion aufdecken.	
	Anmerkung: Ziemlich ähnlich wie Frage 4, jedoch speziell auf Auftrag bezogen und daher etwas anders formuliert.	

	Frage 8	Offen
	Erinnern Sie sich an einen Produktionsauftrag, der „total in die Hose ging“ – was war da passiert? Woran lag das?	
H8.1 H8.2	Antwort - Wenn einzelne individuelle Fehler machen, falsch eingestellt an Maschine - Termintreue (siehe unten 1)	
	Konkretisierungsfrage Woran lag das?	
H8.3 H8.4 H8.5 H8.6	Antwort Konkretisierung - 1 wegen zu vieler Aufträge - Befragung <i>der Kunden ergab, dass qualitativ total gut, auch Fehlerrate ist sehr niedrig</i> - <i>Aber: Termintreue schlechter</i>	
	Ziel: Negativbeispiel zur Identifizierung der Probleme nutzen.	

	Frage 9	Offen
	Erinnern Sie sich an einen Produktionsauftrag, der besonders gut lief – was war da passiert? Woran lag das?	
H9.1	Antwort Ablauf war geplant, lief gut durch, kein „individueller Fehler“, Termin gehalten	
	Ziel: Positivbeispiel zur Identifizierung von Problemen nutzen.	

Frage 10			Bewertung, Offen
Wie wichtig sind die folgenden Aspekte für Sie, bewerten von 1 (unwichtig) bis 5 (sehr wichtig)?			
H10.1	A	Niedrige Durchlaufzeiten der Aufträge	1 2 <u>3</u> 4 5
H10.2	B	Hohe Lieferfähigkeit (inwieweit es dem Vertrieb möglich ist, vom Kunden gewünschte Liefertermine unter Beachtung der Produktionsgegebenheiten zusagen zu können)	1 2 3 4 <u>5</u>
H10.3	C	Hohe Liefertreue (in welchem Maße die bei der Auftragserteilung zugesagten Termine realisiert werden können)	1 2 3 4 <u>5</u>
H10.4	D	Hohe Auslastung der Mitarbeiter	1 2 3 <u>4</u> 5
H10.5	E	Gleichmäßige Auslastung Mitarbeiter (nicht stark schwankend)	1 2 3 4 <u>5</u>
H10.6	F	Hohe Auslastung der Maschinen	1 2 3 <u>4</u> 5
H10.7	G	Gleichmäßige Auslastung Maschinen (nicht stark schwankend)	1 2 3 <u>4</u> 5
H10.8	H	Lagerbestände auf ein möglichst niedriges Niveau zu reduzieren	1 <u>2</u> 3 4 5
H10.9	I	Umlaufbestände auf ein möglichst niedriges Niveau zu reduzieren (z.B. unfertige Erzeugnisse)	1 <u>2</u> 3 4 5
H10.10	<u>K</u>	Bestand Rohmaterial optimieren (reduzieren aber trotzdem zum richtigen Zeitpunkt verfügbar haben)	1 2 <u>3</u> 4 5
H10.11	L	Bestand Hilfsmaterial optimieren (reduzieren aber trotzdem zum richtigen Zeitpunkt verfügbar haben)	1 2 <u>3</u> 4 5
H10.12	M	Bestand fertige Erzeugnisse optimieren (reduzieren aber trotzdem zum richtigen Zeitpunkt verfügbar haben)	nicht relevant
H10.13	N	Material-Warteschlangen reduzieren	1 <u>2</u> 3 4 5
H10.14	O	Flaschenhalse im Produktionsprozess identifizieren	1 2 3 <u>4</u> 5
H10.15	<u>P</u>	Auftragsreihenfolgen planen, um effizienter abzuwickeln	1 2 3 <u>4</u> 5
H10.16	R	Puffergrößen optimieren (Lager zwischen den einzelnen Maschinen)	1 2 <u>3</u> 4 5
H10.17	S	Besser vorher wissen, wann welche Maschine belegt ist	1 2 <u>3</u> 4 5
H10.18	T	Besser vorher wissen, wann welcher Mitarbeiter belegt ist	1 <u>2</u> 3 4 5
H10.19	U	Produktionsprozess anschaulich abbilden	1 2 3 <u>4</u> 5
H10.20	W	Was noch? Termingerechtigkeit	1 2 3 4 <u>5</u>
Antworten Hinterfragungen			

H10.21	K: Zinsen sind niedrig, daher egal
H10.22	P: <i>Machen sie schon mit Access</i>
	Ziel: Gewichtung der einzelnen Zielgrößen hinterfragen.
	Anmerkung: Hilfstafel hinlegen. Markante Antworten jeweils hinterfragen, warum das so gesehen wird.

		Frage 11	Bewertung, Offen
		Kennen/messen Sie folgende Zielgrößen in Ihrem Unternehmen, wenn ja wie oft durchschnittlich (täglich, wöchentlich, monatlich, jährlich, nie)?	
H11.1	A	Ausbringungsmenge	Tag <u>Woche</u> Monat Jahr <u>nie</u>
H11.2	<u>B</u>	Bestand Rohmaterial	Tag <u>Woche</u> Monat Jahr nie
H11.3	<u>C</u>	Bestand Hilfsmaterial	Tag <u>Woche</u> Monat Jahr nie
H11.4	D	Bestand fertige Erzeugnisse	<u>Tag</u> Woche Monat Jahr nie
H11.5	<u>E</u>	Durchlaufzeit der Aufträge	Tag Woche Monat Jahr <u>nie</u>
H11.6	<u>F</u>	Auslastungsgrad Maschinen	Tag Woche Monat Jahr <u>nie</u>
H11.7	G	Auslastungsgrad Mitarbeiter	Tag Woche Monat Jahr <u>nie</u>
H11.8	H	Termintreue bei Produktionsaufträgen	Tag Woche Monat Jahr <u>nie</u>
H11.9	I	Lagerverweilzeit fertige Erzeugnisse (wie lange bleiben fertige Erzeugnisse im Lager)	nicht zutreffend
		Antworten Hinterfragungen	
H11.10	B: <i>Programm + Sichtprüfung ist</i>		
H11.11	C: <i>Sichtprüfung ist</i>		
H11.12	E: Nicht interessant (nachher)		
H11.13	F: Wird nicht gemessen, nach Gefühl		
		Ziel: Feststellen, welche Kennzahlen vorhanden sind.	
		Anmerkung: Hilfstafel hinlegen. Antworten ggf. hinterfragen, z.B. wie das gemessen wird oder warum nicht gemessen wird. Im nächsten Schritt anschauen.	

	Frage 12	Offen
	Welches sind die wichtigsten Maschinen in Ihrem Unternehmen?	
H12.1	Antwort Jede gleich wichtig	
	Konkretisierungsfrage Wenn welche Maschine defekt wäre, wo wäre das das größte Problem?	
H12.2 H12.3	Antwort Konkretisierung Manche können im Produktionsprozess nicht <i>ausgetauscht</i> werden, z.B. große Brother II für Einlegeteile	
	Ziel: Flaschenhalse herausfinden.	

	Frage 13	Offen
	Welche Maschinen hat ein Unternehmen Ihrer Branche und Größe typischerweise?	
H13.1 H13.2	Antwort - Alles CNC: Fräsmaschinen, Drehmaschinen, Dreh-/Fräszentren - Gleitschleifmaschinen	
	Ziel: Herausfinden welche Maschinen modelliert werden müssen.	

Bereich Prozess-Simulation

	Frage 14	Offen
	Was haben Sie bisher von Prozess-Simulation gehört?	
H14.1 H14.2 H14.3	Antwort - PPS-MDE - Mehr Aufwand als Nutzen - Muss viel eingegeben werden	
	Ziel: Hintergrund dieses Unternehmens bezüglich Prozess-Simulation erfragen.	

	Frage 15	Offen
	Haben Sie in Ihrem Unternehmen schon mal etwas mit Prozess-Simulation gemacht?	
H15.1	Antwort <i>Nein</i>	
	Konkretisierungsfrage Warum besteht da kein Interesse bei Ihnen?	
H15.2 H15.3	Antwort Konkretisierung <i>Muss man viel eingeben, dann kommt was ganz genaues raus, das auf ungenauen Zahlen basiert und nie mit der Realität übereinstimmt</i>	
	Ziel: Hintergrund dieses Unternehmens bezüglich Prozess-Simulation erfragen.	

	Frage 16	Offen
	Was stellen Sie sich vor, was die Prozess-Simulation leisten kann?	
H16.1	Antwort Termin genau zusagen, das auch einhalten	
	Ziel: Einschätzung bekommen, wie Interviewpartner den Nutzen von Prozess-Simulation einschätzt.	

	Frage 17	Offen
	In welchem Bereich könnten Sie sich die Anwendung von Prozess-Simulation bei Ihnen am ehesten vorstellen?	
H17.1	Antwort In der Fertigung	
	Ziel: Einschätzung bekommen, wo Akzeptanz für Prozess-Simulation gegeben wäre.	

	Frage 18	Offen
	Welche Bedenken haben Sie im Bezug auf Prozess-Simulation?	
	Antwort	
H18.1	- Genaues Erfassen von „geschätzten Daten“	
H18.2	- Unvorhersehbare „Zusatzaufträge“, z.B. Anruf, wenn <i>schon auf Maschine</i>	
H18.3	→ <i>kann Programm gar nicht</i>	
H18.4	- Kann „Bauchentscheidungen“ nicht toppen	
H18.5	- Flexibilität geht verloren, wenn ich erst alles eingeben muss	
H18.6	- Wir wollen flexibel bleiben	
	Konkretisierungsfrage	
	Was denken Sie für wen der Einsatz von Prozess-Simulation sinnvoll wäre?	
	Antwort Konkretisierung	
H18.7	<i>Sinnvoll bei Unternehmen so ab 50 Leuten (-100) dann eine Person dafür abstellen</i>	
	Ziel: Einschätzung bezüglich Bedenken bekommen.	

Weitere Anmerkungen:

HWA1	- Prozess-Simulation interessant ab ca. 40-50 Leuten, das ist so eine „kritische
HWA2	Größe“, wo man nicht mehr alles im Kopf hat
HWA3	- Alles CNC
HWA4	- Hilfsmittel Access: Bestellungen nach Liefertermin nach Wert, je Monat
HWA5	- Datenerfassung nicht so genau
HWA6	- <i>Sehr eilige Aufträge werden auch schon mal dazwischen geschoben, Rüstzeit</i>
HWA7	<i>dann „egal“</i>
HWA8	- Früher Auftragsliste für alle
HWA9	- Auftragsliste nach Kunde, Liefertermin
HWA10	- Idee: Lager über Barcode + Tablet scannen
HWA11	- <i>Nutzt ein selbst programmiertes Access-Werkzeug, gibt dort Auftrag ein,</i>
HWA12	<i>errechnet Material, Termin wann fertig ist, welcher Auftrag auf welcher</i>
HWA13	<i>Maschine läuft</i>
HWA14	- <i>Staffelpreise</i>
HWA15	- <i>Materialbeschaffung</i>
HWA16	- <i>Access-Lösung geht in Richtung Plantafel</i>

Protokoll Besichtigung Produktion

HPB1	- <i>Halle wirkt durch den fehlenden Platz „chaotisch“, kein Überblick was wo läuft,</i>
HPB2	<i>überall Lager, verwinkelt</i>
HPB3	- <i>An den Maschinen hängen teilweise die Abarbeitungslisten</i>
HPB4	- <i>Es wird meist mit Stangenprofilen gearbeitet</i>
HPB5	- <i>Möglichst viele Maschinen auf engem Raum, überlappen sich teilweise</i>
HPB6	- <i>Gutes Klima bei der Belegschaft</i>
HPB7	- <i>Im Keller weitere Maschinen, ein Materiallager, man muss die Treppe runter</i>
HPB8	- <i>Im Keller kaum jemand am arbeiten</i>
HPB9	- <i>Macht viel „aus dem Bauch“, z.B. auch Materialverbrauch planen (mit cm-Maß</i>
HPB10	<i>und dann rechnen)</i>
HPB11	- <i>Im Keller Geitschleifmaschine</i>

Anhang A3

Experteninterview Zilske

Leitfadeninterview

Nachträglich erfasstes ist *kursiv* dargestellt

Rahmendaten

Unternehmen	A. Zielske, Präzisionsmechanik GmbH
Anzahl Mitarbeiter im Unternehmen	2 + einige Aushilfen (z.B. für die Messmaschine)
Standort Unternehmen	Hochstraße 37, 82024 Taufkirchen
Herstellung von Metallernzeugnissen	<u>Ja</u> Nein
Ort, Datum Interview	Taufkirchen, 27.3.2014
Name Interviewpartner 1	Herr Zielske
Funktion Interviewpartner 1	Geschäftsführer
Name Interviewpartner 2	Herr Schäfer
Funktion Interviewpartner 2	Werksleitung

Bereich Produktion & Logistik

	Frage 1	Offen
	Bitte erzählen Sie mir erst einmal was für Metallerzeugnisse Ihr Unternehmen herstellt.	
Z1.1 Z1.2 Z1.3	Antwort - Lohnfertigung nach Zeichnung + Auftrag (Datei) (wenige Dinge werden mitverbaut), viel von Siemens, alte Beziehungen, „Mundpropaganda“ - Feinmechanik	
	Konkretisierungsfrage Was ist ein typischer Auftrag, den Sie bekommen?	
Z1.4 Z1.5 Z1.6	Antwort Konkretisierung - Stückzahlen 2-10 Stück - Siemens „Dauerläufer“ im Jahr ca. 500-800 - Hohe Qualität, Flexibilität	
	Ziel: Offene Einstiegsfrage. Generelle Erzeugnisse kennenlernen.	

	Frage 2	Offen
	Wie läuft denn ein typischer Auftrag durch Ihre Produktion – vom Auftragsingang bis zur Auslieferung?	
Z2.1 Z2.2 Z2.3 Z2.4 Z2.5 Z2.6 Z2.7 Z2.8 Z2.9 Z2.10	Antwort - Kunde mailt Daten als Zeichnung oder Volumenmodell - Angebot+Terminzusage (Termin wird später teils/teils gehalten) - Bestellung - Lösungsweg suchen, programmieren - Material bestellen - Bearbeiten - Evtl. extern zur Weiterbearbeitung geben - Evtl. Prüfprotokoll erstellen - Verpacken - Lieferung	
Z2.11	Konkretisierung Nutzen digitale Zeiterfassung über kleine Geräte, wird von ISO9000 gefordert	
	Ziel: Offene Einstiegsfrage. Generellen Ablauf kennenlernen.	

	Frage 3	Geschlossen
	Produzieren Sie nur „auf Auftrag“?	
Z3.1	Antwort Ja	
	Konkretisierungsfrage Oder auch auf Vorrat?	
Z3.2	Antwort Konkretisierung Evtl. kleine Mengen wenn man weiß dass der wiederkommt	
	Ziel: Herausfinden ob über die Produktion selbst bestimmt werden kann oder ausschließlich vom Kunden getrieben wird.	

	Frage 4	Offen
	Wo sehen Sie das größte Optimierungspotential Ihrer Produktion und Logistik?	
Z4.1 Z4.2	Antwort CNC-Software pflegen auf neuesten Stand (neue Fräs-Strategien, muss man umstellen)	
	Konkretisierungsfrage Wo könnten man am ehesten etwas verbessern?	
Z4.3	Antwort Konkretisierung Ginge nur mit Umziehen (wachsen)	
	Ziel: Einschätzung eines Experten bekommen, wo Potential zur Optimierung gesehen wird.	

	Frage 5	Offen
	Was sind die größten Probleme in Ihrer Produktion?	
Z5.1	Antwort <i>Keine</i>	
	Ziel: Probleme in der Produktion herausfinden.	

	Frage 6	Offen
	Was sind die größten Probleme in Ihrer Logistik?	
Z6.1	Antwort <i>Keine</i>	
	Ziel: Probleme in der Produktionslogistik herausfinden.	
	Anmerkung: Selbst aussortieren, was zur Produktionslogistik gehört und was nicht. Interviewpartner sollte das nicht müssen.	

	Frage 7	Offen
	Was macht bei Produktionsaufträgen die meisten Schwierigkeiten?	
Z7.1	Antwort Siehe vorne, Frage 4	
	Ziel: Schwierigkeiten in der Produktion aufdecken.	
	Anmerkung: Ziemlich ähnlich wie Frage 4, jedoch speziell auf Auftrag bezogen und daher etwas anders formuliert.	

	Frage 8	Offen
	Erinnern Sie sich an einen Produktionsauftrag, der „total in die Hose ging“ – was war da passiert? Woran lag das?	
Z8.1 Z8.2	Antwort Ist lange her, falsche technische Einschätzung der Umgebung in einem Salzbergwerk (Material korrodierte)	
	Ziel: Negativbeispiel zur Identifizierung der Probleme nutzen.	

	Frage 9	Offen
	Erinnern Sie sich an einen Produktionsauftrag, der besonders gut lief – was war da passiert? Woran lag das?	
Z9.1	Antwort Nichts herausragendes	
	Ziel: Positivbeispiel zur Identifizierung von Problemen nutzen.	

Frage 10			Bewertung, Offen
Wie wichtig sind die folgenden Aspekte für Sie, bewerten von 1 (unwichtig) bis 5 (sehr wichtig)?			
Z10.1	A	Niedrige Durchlaufzeiten der Aufträge	1 2 3 <u>4</u> 5
Z10.2	B	Hohe Lieferfähigkeit (inwieweit es dem Vertrieb möglich ist, vom Kunden gewünschte Liefertermine unter Beachtung der Produktionsgegebenheiten zusagen zu können)	1 2 3 <u>4</u> 5
Z10.3	C	Hohe Liefertreue (in welchem Maße die bei der Auftragserteilung zugesagten Termine realisiert werden können)	1 2 3 <u>4</u> 5
Z10.4	D	Hohe Auslastung der Mitarbeiter	1 2 3 <u>4</u> 5
Z10.5	E	Gleichmäßige Auslastung Mitarbeiter (nicht stark schwankend)	1 2 <u>3</u> 4 5
Z10.6	<u>F</u>	Hohe Auslastung der Maschinen	1 2 <u>3</u> 4 5
Z10.7	G	Gleichmäßige Auslastung Maschinen (nicht stark schwankend)	1 <u>2</u> 3 4 5
Z10.8	<u>H</u>	Lagerbestände auf ein möglichst niedriges Niveau zu reduzieren	<i>siehe unten</i>
Z10.9	I	Umlaufbestände auf ein möglichst niedriges Niveau zu reduzieren (z.B. unfertige Erzeugnisse)	1 2 3 <u>4</u> 5
Z10.10	K	Bestand Rohmaterial optimieren (reduzieren aber trotzdem zum richtigen Zeitpunkt verfügbar haben)	1 2 3 <u>4</u> 5
Z10.11	L	Bestand Hilfsmaterial optimieren (reduzieren aber trotzdem zum richtigen Zeitpunkt verfügbar haben)	1 <u>2</u> 3 4 5
Z10.12	<u>M</u>	Bestand fertige Erzeugnisse optimieren (reduzieren aber trotzdem zum richtigen Zeitpunkt verfügbar haben)	<i>keine fertigen Erzeugnisse</i>
Z10.13	N	Material-Warteschlangen reduzieren	1 <u>2</u> 3 4 5
Z10.14	O	Flaschenhalse im Produktionsprozess identifizieren	1 <u>2</u> 3 4 5
Z10.15	P	Auftragsreihenfolgen planen, um effizienter abzuwickeln	1 2 3 4 <u>5</u>
Z10.16	R	Puffergrößen optimieren (Lager zwischen den einzelnen Maschinen)	keine vorhanden
Z10.17	S	Besser vorher wissen, wann welche Maschine belegt ist	1 2 3 4 <u>5</u>
Z10.18	T	Besser vorher wissen, wann welcher Mitarbeiter belegt ist	1 <u>2</u> 3 4 5
Z10.19	U	Produktionsprozess anschaulich abbilden	<u>1</u> 2 3 4 5
Antworten Hinterfragungen			
Z10.20	F: Kommt auf Abbezahlung an		

Z10.21	H: Fertige Erzeugnisse schon (siehe vorne), Verbrauchsmaterial egal
Z10.22	M: Keine fertigen Erzeugnisse
	Ziel: Gewichtung der einzelnen Zielgrößen hinterfragen.
	Anmerkung: Hilfstafel hinlegen. Markante Antworten jeweils hinterfragen, warum das so gesehen wird.

Frage 11			Bewertung, Offen
Kennen/messen Sie folgende Zielgrößen in Ihrem Unternehmen, wenn ja wie oft durchschnittlich (täglich, wöchentlich, monatlich, jährlich, nie)?			
Z11.1	A	Ausbringungsmenge	<u>Tag</u> Woche Monat Jahr <u>nie</u>
Z11.2	B	Bestand Rohmaterial	<u>Tag</u> Woche Monat Jahr <u>nie</u>
Z11.3	C	Bestand Hilfsmaterial	Tag Woche <u>Monat</u> Jahr <u>nie</u>
Z11.4	D	Bestand fertige Erzeugnisse	nicht zutreffend
Z11.5	E	Durchlaufzeit der Aufträge	unregelmäßig
Z11.6	F	Auslastungsgrad Maschinen	Tag Woche Monat Jahr <u>nie</u>
Z11.7	G	Auslastungsgrad Mitarbeiter	Tag Woche Monat Jahr <u>nie</u>
Z11.8	H	Termintreue bei Produktionsaufträgen	nach Bedarf
Z11.9	I	Lagerverweilzeit fertige Erzeugnisse (wie lange bleiben fertige Erzeugnisse im Lager)	Tag Woche Monat Jahr <u>nie</u>
Antworten Hinterfragungen			
Z11.10	E: Unregelmäßig, nach Bedarf		
Z11.11	H: Nach Bedarf, wie verschiebt sich das		
Ziel: Feststellen, welche Kennzahlen vorhanden sind.			
Anmerkung: Hilfstafel hinlegen. Antworten ggf. hinterfragen, z.B. wie das gemessen wird oder warum nicht gemessen wird. Im nächsten Schritt anschauen.			

Frage 12		Offen
Welches sind die wichtigsten Maschinen in Ihrem Unternehmen?		
Z12.1	Antwort 3 CNC-Fräsmaschinen	
Konkretisierungsfrage Warum sind die für Sie wichtig?		
Z12.2	Antwort Konkretisierung Damit kann man am schnellsten und einfachsten produzieren	
Ziel: Herausfinden welche Maschinen modelliert werden müssen.		

	Frage 13	Offen
	Welche Maschinen hat ein Unternehmen Ihrer Branche und Größe typischerweise?	
Z13.1	Antwort	
Z13.2	CNC-Drehmaschinen, manuelle Dreh-/Fräsen	
	Kombinierte ab ca. 10 Leuten	
	Ziel: Herausfinden welche Maschinen modelliert werden müssen.	

Bereich Prozess-Simulation

	Frage 14	Offen
	Was haben Sie bisher von Prozess-Simulation gehört?	
Z14.1	Antwort Abläufe simulieren in Planung DLZ, Endergebniss	
	Ziel: Hintergrund dieses Unternehmens bezüglich Prozess-Simulation erfragen.	

	Frage 15	Offen
	Haben Sie in Ihrem Unternehmen schon mal etwas mit Prozess-Simulation gemacht?	
Z15.1	Antwort Nein	
	Konkretisierungsfrage Warum besteht da kein Interesse bei Ihnen?	
Z15.2	Antwort Konkretisierung Mit Daten füttern, pflegen, zu großer Aufwand	
	Ziel: Hintergrund dieses Unternehmens bezüglich Prozess-Simulation erfragen.	

	Frage 16	Offen
	Was stellen Sie sich vor, was die Prozess-Simulation leisten kann?	
Z16.1	Antwort Sorgfältige Planung, Personal + Betriebsmittel	
	Ziel: Einschätzung bekommen, wie Interviewpartner den Nutzen von Prozess-Simulation einschätzt.	

	Frage 17	Offen
	In welchem Bereich könnten Sie sich die Anwendung von Prozess-Simulation bei Ihnen am ehesten vorstellen?	
Z17.1	Antwort Auftragsablauf	
	Ziel: Einschätzung bekommen, wo Akzeptanz für Prozess-Simulation gegeben wäre.	

	Frage 18	Offen
	Welche Bedenken haben Sie im Bezug auf Prozess-Simulation?	
Z18.1	Antwort Viel Aufwand für Eingabe+Pflege, lohnt sich Aufwand? Zeitaufwand wichtiger als Anschaffungskosten	
	Ziel: Einschätzung bezüglich Bedenken bekommen.	

Weitere Anmerkungen:

ZWA1	- Zeitaufwand wichtiger als Anschaffungskosten
ZWA2	- Erst wird Machbarkeit geprüft

Protokoll Besichtigung Produktion

ZPB1	- <i>Sehr kleine Produktion, übersichtlich</i>
ZPB2	- <i>Gut aufgeräumtes Lager (Rohstoffe + fertige Erzeugnisse)</i>
ZPB3	- <i>Extra Prüfraum</i>

Anhang A4

Experteninterview Markl

Leitfadeninterview Nachträglich erfasstes ist *kursiv* dargestellt

Rahmendaten

Unternehmen	Feinmechanische Werkstätte Thomas Markl GmbH
Anzahl Mitarbeiter im Unternehmen	24 Mitarbeiter inklusive Aushilfen und Azubis
Standort Unternehmen	Raiffeisenallee 12, 82041 Oberhaching
Herstellung von Metallerzeugnissen	<u>Ja</u> Nein
Ort, Datum Interview	Oberhaching, 22.5.2014
Name Interviewpartner 1	Matthias Schimmel
Funktion Interviewpartner 1	Produktionsleiter, rechte Hand des Geschäftsführers

Bereich Produktion & Logistik

	Frage 1	Offen
	Bitte erzählen Sie mir erst einmal was für Metallerzeugnisse Ihr Unternehmen herstellt.	
M1.1 M1.2 M1.3	Antwort - Dienstleister, Lohnfertigung - Zerspanend, kleine (Uhrenteile) bis 1x1m große Teile - Alu, Stahl, Titan, Inkonel, Messing, Kupfer, faserverstärkte Kunststoffe	
	Konkretisierungsfrage Was ist ein typischer Auftrag, den Sie bekommen?	
M1.4 M1.5	Antwort Konkretisierung - 1-1000 Teile - Auch Raumfahrt (Astrium)	
	Ziel: Offene Einstiegsfrage. Generelle Erzeugnisse kennenlernen.	

	Frage 2	Offen
	Wie läuft denn ein typischer Auftrag durch Ihre Produktion – vom Auftragseingang bis zur Auslieferung?	
M2.1 M2.2 M2.3 M2.4 M2.5 M2.6 M2.7 M2.8 M2.9 M2.10 M2.11 M2.12 M2.13 M2.14	Antwort - <i>Anfrage</i> : Modell (3D) und Zeichnung/Worte - Prüfen ob geht - Kalkulation - Angebotserstellung (<i>mit</i> Bios2000 ERP-System) - Auftrag - Arbeitsplan → Fertigungsauftrag - Materialbeschaffung - Fertigungspapiere (Bios2000: Arbeitskarten + Zeichnung) - Leitstand-Planung geplant - Programmierung (aufwändige Teile, <i>braucht</i> viele <i>verschiedene</i> Maschinen) - Fertigung - (Prüfung während Fertigung) - Evtl. Prüfprotokoll - Verpacken	
	Konkretisierungsfrage Werden die Termine schon zum Angebot zugesagt, und wie werden diese bestimmt?	
M2.15 M2.16	Antwort Konkretisierung Termine <i>werden geschätzt</i> , verschieben sich; Bestimmung über Plantafel, <i>wird aber kaum genutzt</i>	
	Ziel: Offene Einstiegsfrage. Generellen Ablauf kennenlernen.	

	Frage 3	Geschlossen
	Produzieren Sie nur „auf Auftrag“?	
M3.1	Antwort <i>Ja, nur Lohnfertigung</i>	
	Ziel: Herausfinden ob über die Produktion selbst bestimmt werden kann oder ausschließlich vom Kunden getrieben wird.	

	Frage 4	Offen
	Wo sehen Sie das größte Optimierungspotential Ihrer Produktion und Logistik?	
M4.1 M4.2 M4.3 M4.4	Antwort - Plantafel hat Zusatzmodule, aber viele Infos „im Kopf“, z.B. Fähigkeiten Mitarbeiter, Urlaubsplanung, Krankheiten (<i>extrahieren in Programm?</i>) - Weniger Werkzeuge auf Vorrat (Wert!) - Logistik „müsste man aufräumen“	
	Konkretisierungsfrage Wo könnten man am ehesten etwas verbessern?	
M4.5	Antwort Konkretisierung Plantafel plant keine Mitarbeiter	
	Konkretisierungsfrage Welcher Bereich ist momentan am „schlechtesten“?	
M4.6	Antwort Konkretisierung CNC-Programmierung, wir brauchen zu lang → eingesetztes Programm komplex	
	Ziel: Einschätzung eines Experten bekommen, wo Potential zur Optimierung gesehen wird.	

	Frage 5	Offen
	Was sind die größten Probleme in Ihrer Produktion?	
M5.1 M5.2 M5.3 M5.4	Antwort - technische Probleme - Material „unkalkulierbar“ - Kollegen krank - Freitags Fertigungsplanung → Montag ändert sich weil krank, <i>Zusatzaufträge, ...</i>	
	Konkretisierungsfrage Wo gibt es immer wieder Probleme?	
M5.5 M5.6	Antwort Konkretisierung Wenn 4 statt 3 Tage braucht (<i>wie vorher geplant</i>), verschiebt sich Alles weil mehrere Maschinen belegt	
	Ziel: Probleme in der Produktion herausfinden.	

	Frage 6	Offen
	Was sind die größten Probleme in Ihrer Logistik?	
M6.1 M6.2	Antwort - Keine - Macht alles eine Person (<i>Hr. Schimmel</i>), Erfahrungswerte „im Kopf“	
	Ziel: Probleme in der Produktionslogistik herausfinden.	
	Anmerkung: Selbst aussortieren, was zur Produktionslogistik gehört und was nicht. Interviewpartner sollte das nicht müssen.	

	Frage 7	Offen
	Was macht bei Produktionsaufträgen die meisten Schwierigkeiten?	
M7.1	Antwort Siehe vorne, <i>nur Lohnfertigung, daher Frage identisch mit 4</i>	
	Ziel: Schwierigkeiten in der Produktion aufdecken.	
	Anmerkung: Ziemlich ähnlich wie Frage 4, jedoch speziell auf Auftrag bezogen und daher etwas anders formuliert.	

	Frage 8	Offen
	Erinnern Sie sich an einen Produktionsauftrag, der „total in die Hose ging“ – was war da passiert? Woran lag das?	
M8.1 M8.2 M8.3 M8.4	Antwort <i>Das was momentan läuft!</i> - Kunde: geschoben, Teile getauscht, neue Zeichnung die verkehrt war - Kollegen: Werkzeug falsch gewählt - Material: unkalkulierbar <i>Inkonel</i> (für Triebwerke)	
	Ziel: Negativbeispiel zur Identifizierung der Probleme nutzen.	

	Frage 9	Offen
	Erinnern Sie sich an einen Produktionsauftrag, der besonders gut lief – was war da passiert? Woran lag das?	
M9.1 M9.2	Antwort - CNC-Programmierung gut - Werkzeugwahl richtig, eingestellt, eingemessen, <i>simuliert</i>	
	Konkretisierungsfrage Gibt es Aufträge die viel mehr Gewinn bringen als vorher gedacht?	
M9.3	Antwort Konkretisierung Wenn von Erfahrungen von anderen Aufträgen profitiert werden kann	
	Konkretisierungsfrage Wie ist denn das Verhältnis neue und Folgeaufträge?	
M9.4 M9.5	Antwort Konkretisierung 40% neue, 60% Folgeaufträge <i>auch bei Folgeaufträgen wird CNC-programmiert (Optimierung!)</i>	
	Ziel: Positivbeispiel zur Identifizierung von Problemen nutzen.	

Frage 10			Bewertung, Offen
Wie wichtig sind die folgenden Aspekte für Sie, bewerten von 1 (unwichtig) bis 5 (sehr wichtig)?			
M10.1	<u>A</u>	Niedrige Durchlaufzeiten der Aufträge	1 2 3 4 <u>5</u>
M10.2	B	Hohe Lieferfähigkeit (inwieweit es dem Vertrieb möglich ist, vom Kunden gewünschte Liefertermine unter Beachtung der Produktionsgegebenheiten zusagen zu können)	1 2 3 <u>4</u> 5
M10.3	C	Hohe Liefertreue (in welchem Maße die bei der Auftragserteilung zugesagten Termine realisiert werden können)	1 2 3 <u>4</u> 5
M10.4	D	Hohe Auslastung der Mitarbeiter	1 2 3 4 <u>5</u>
M10.5	E	Gleichmäßige Auslastung Mitarbeiter (nicht stark schwankend)	1 2 3 4 <u>5</u>
M10.6	F	Hohe Auslastung der Maschinen	1 2 3 <u>4</u> 5
M10.7	G	Gleichmäßige Auslastung Maschinen (nicht stark schwankend)	1 2 3 <u>4</u> 5
M10.8	H	Lagerbestände auf ein möglichst niedriges Niveau zu reduzieren	1 2 3 <u>4</u> 5
M10.9	I	Umlaufbestände auf ein möglichst niedriges Niveau zu reduzieren (z.B. unfertige Erzeugnisse)	1 2 3 <u>4</u> 5
M10.10	<u>K</u>	Bestand Rohmaterial optimieren (reduzieren aber trotzdem zum richtigen Zeitpunkt verfügbar haben)	1 2 3 <u>4</u> 5
M10.11	L	Bestand Hilfsmaterial optimieren (reduzieren aber trotzdem zum richtigen Zeitpunkt verfügbar haben)	1 2 <u>3</u> 4 5
M10.12	M	Bestand fertige Erzeugnisse optimieren (reduzieren aber trotzdem zum richtigen Zeitpunkt verfügbar haben)	<i>nur Lohnfertigung</i>
M10.13	N	Material-Warteschlangen reduzieren	1 2 3 <u>4</u> 5
M10.14	<u>Q</u>	Flaschenhalse im Produktionsprozess identifizieren	1 2 3 <u>4</u> 5
M10.15	<u>P</u>	Auftragsreihenfolgen planen, um effizienter abzuwickeln	1 2 3 4 <u>5</u>
M10.16	R	Puffergrößen optimieren (Lager zwischen den einzelnen Maschinen)	1 2 3 <u>4</u> 5
M10.17	<u>S</u>	Besser vorher wissen, wann welche Maschine belegt ist	1 <u>2</u> 3 4 5
M10.18	T	Besser vorher wissen, wann welcher Mitarbeiter belegt ist	1 <u>2</u> 3 4 5
M10.19	<u>U</u>	Produktionsprozess anschaulich abbilden	<u>1</u> 2 3 4 5
Antworten Hinterfragungen			
M10.20	A: Pauschal schwierig, eher „optimale Durchlaufzeit“ Teile brauchen je nach		

M10.21	<i>Komplexität länger/kürzer</i>
M10.22	K: Teilweise Vorrat, teilweise bestellt
M10.23	O: Kennen sie, wäre teuer (<i>neue, zusätzliche Maschine</i>)
M10.24	P: Plantafel <i>macht</i>
M10.25	S: <i>Plantafel, wir wissen schon</i> , Aufträge da Oktober, noch nicht auf Maschine geplant U: Plantafel macht schon
	Ziel: Gewichtung der einzelnen Zielgrößen hinterfragen.
	Anmerkung: Hilfstafel hinlegen. Markante Antworten jeweils hinterfragen, warum das so gesehen wird.

	Frage 11	Bewertung, Offen
	Kennen/messen Sie folgende Zielgrößen in Ihrem Unternehmen, wenn ja wie oft durchschnittlich (täglich, wöchentlich, monatlich, jährlich, nie)?	
M11.1	<u>A</u> Ausbringungsmenge	Tag <u>Woche</u> Monat Jahr <u>nie</u>
M11.2	B Bestand Rohmaterial	Tag Woche Monat Jahr <u>nie</u>
M11.3	<u>C</u> Bestand Hilfsmaterial	Tag Woche Monat Jahr <u>nie</u>
M11.4	D Bestand fertige Erzeugnisse	<i>nicht relevant</i>
M11.5	E Durchlaufzeit der Aufträge	Tag Woche Monat Jahr <u>nie</u>
M11.6	F Auslastungsgrad Maschinen	<u>Tag</u> Woche Monat Jahr <u>nie</u>
M11.7	<u>G</u> Auslastungsgrad Mitarbeiter	<u>Tag</u> Woche Monat Jahr <u>nie</u>
M11.8	<u>H</u> Termintreue bei Produktionsaufträgen	<u>Tag</u> Woche Monat Jahr <u>nie</u>
M11.9	I Lagerverweilzeit fertige Erzeugnisse (wie lange bleiben fertige Erzeugnisse im Lager)	nicht zutreffend
	Antworten Hinterfragungen	
M11.10	A: Schwankt stark wegen unterschiedlicher Teile, <i>über Rechnung</i>	
M11.11	C: Z.B. Öle, Kühlmittel, Werkzeuge	
M11.12	G: Wenn Plantafel „voll“ Mitarbeiter auch „voll“	
M11.13	H: Könnte System, aber wird Termin nicht gepflegt; nicht dokumentiert; täglich mit Lieferschein („ <i>sehe ich ja dann</i> “)	
	Ziel: Feststellen, welche Kennzahlen vorhanden sind.	
	Anmerkung: Hilfstafel hinlegen. Antworten ggf. hinterfragen, z.B. wie das gemessen wird oder warum nicht gemessen wird. Im nächsten Schritt anschauen.	

	Frage 12	Offen
	Welches sind die wichtigsten Maschinen in Ihrem Unternehmen?	
M12.1	Antwort 5-Achs-Fräsmaschine CNC (teuerste) 5x <i>vorhanden</i>	
	Ziel: Flaschenhäse herausfinden.	

	Frage 13	Offen
	Welche Maschinen hat ein Unternehmen Ihrer Branche und Größe typischerweise?	
M13.1	Antwort - Fräsmaschine, Drehmaschine (CNC <i>und</i> konventionell)	
M13.2	- Schleifmaschine, Bohrmaschine (konventionell)	
M13.3	- Gleitschleif-Maschine (<i>Trowalisieren</i>) zum <i>Entgraten</i> („ <i>Waschmaschine</i> “)	
	Ziel: Herausfinden welche Maschinen modelliert werden müssen.	

Bereich Prozess-Simulation

	Frage 14	Offen
	Was haben Sie bisher von Prozess-Simulation gehört?	
M14.1	Antwort Nur von Fr. Rudel	
	Ziel: Hintergrund dieses Unternehmens bezüglich Prozess-Simulation erfragen.	

	Frage 15	Offen
	Haben Sie in Ihrem Unternehmen schon mal etwas mit Prozess-Simulation gemacht?	
M15.1	Antwort Nein, nur Plantafel	
	Konkretisierungsfrage Warum besteht da kein Interesse bei Ihnen?	
M15.2	Antwort Konkretisierung - Bisher kein Anlass	
M15.3	- In Zeitschriften nix drin (<i>sie lesen</i> VDI-Nachrichten: Produktionsblätter sowie die	
M15.4	„Produktion“)	
	Ziel: Hintergrund dieses Unternehmens bezüglich Prozess-Simulation erfragen.	

	Frage 16	Offen
	Was stellen Sie sich vor, was die Prozess-Simulation leisten kann?	
M16.1	Antwort Bisher nichts vorstellbar, wertschöpfender Prozess wird mit Plantafel simuliert	
	Ziel: Einschätzung bekommen, wie Interviewpartner den Nutzen von Prozess-Simulation einschätzt.	

	Frage 17	Offen
	In welchem Bereich könnten Sie sich die Anwendung von Prozess-Simulation bei Ihnen am ehesten vorstellen?	
M17.1	Antwort Produktion? Vorgelagert/Nachgelagert nicht vorstellbar	
	Ziel: Einschätzung bekommen, wo Akzeptanz für Prozess-Simulation gegeben wäre.	

	Frage 18	Offen
	Welche Bedenken haben Sie im Bezug auf Prozess-Simulation?	
M18.1	Antwort Keine	
	Ziel: Einschätzung bezüglich Bedenken bekommen.	

Weitere Anmerkungen:

MWA1	- Schnittstellen sind das Problem (viel IT, untypisch für diese Größe),
MWA2	verschiedene Software, wenn was geändert wird müssen Schnittstellen angepasst
MWA3	werden
MWA4	- Update + Software viele Tausend Euro im Jahr, ca. 40 TEUR
MWA5	- ERP kostet ca. 20 TEUR („nicht so teuer“), zusätzliche Module wie z.B.
MWA6	Zeitverwaltung „teuer“
MWA7	- IT-Software für ca. 400 TEUR im Betrieb
MWA8	- Plantafel Bios2000 vorgeführt
MWA9	- Jede Maschine einzelne Zeile
MWA10	- Kann man „Maschinenarbeitszeiten“ eingeben (z.B. 8 std., 2 std.)
MWA11	- Sieht aus wie Gantt-Diagramm
MWA12	- Kann man Aufträge hin- und herschieben
MWA13	- Farben zeigen ob zu spät sind (Termin)
MWA14	- Einrüstzeiten werden auch eingeplant, per Hand

Protokoll Besichtigung Produktion

MPB1	- Geräumige Halle, wirkt aufgeräumt
MPB2	- Zentrales „Zwischenlager“ für halbfertige Erzeugnisse
MPB3	- Bei jedem Teil Mappe dabei
MPB4	- Laufkarte, Protokolle, Zeichnung, evtl. Fotos etc.
MPB5	- Einzelne Blätter z.B Fräsen, Schleifen steht jeweils Maschine drauf,
MPB6	Dateipfad; Mitarbeiter NICHT vorgegeben
MPB7	- Viele teure/hochwertige Maschinen, alles wirkt geordnet
MPB8	- Extra Prüfraum, klimatisiert, kann Maschine Prüfprotokoll erstellen
MPB9	- Arbeitsvorbereitung → extra Raum für CNC-Programmierung →
MPB10	CNC-Simulation vorgeführt
MPB11	- Industrie-Spülmaschine mit Ölabscheider zum Waschen von Teilen
MPB12	- Extra Raum für Werkzeuge und Werkzeugvorbereitung (ganzer Schrank voll!)
MPB13	- Würth-Kanban (Schrauben, Scheiben etc.)

Anhang A5

Simulationsmodell Dissertation
(Dokumentation der Autorin der Dissertation)

Dokumentation

Simulationsmodell Dissertation

Autor: Steffi Rudel

Version: 1.0

Datum: 15.01.2016

Inhaltsverzeichnis

Abkürzungs- und Akronymverzeichnis.....	III
Symbolverzeichnis.....	IV
Tabellenverzeichnis.....	V
Abbildungsverzeichnis.....	VI
1 Problemstellung.....	1
2 Rahmenbedingungen.....	2
2.1 Genutzte Software.....	2
2.2 Einschränkungen.....	2
3 Excel.....	3
3.1 Allgemeines.....	3
3.2 Hilfsblätter Anwender.....	4
3.2.1 Anmerkungen.....	4
3.2.2 Umrechnung.....	4
3.3 Eingabe.....	4
3.3.1 Betriebszeiten.....	4
3.3.2 Aufträge.....	5
3.4 Ausgabe.....	6
3.4.1 Ergebnisse_Aufträge.....	6
3.4.2 Ergebnisse_Stationen.....	7
3.5 Hilfsblätter PACE.....	8
3.5.1 PACE_EIN.....	8
3.5.2 PACE_AUS.....	10
3.5.3 PACE_AUS_Masch.....	10
3.6 Hilfsblatt Excel: Eingabewerte.....	11
4 PACE.....	12
4.1 Dokumentation.....	12
4.2 Attribute.....	12
4.3 Variablen.....	13
4.3.1 Globale Variablen.....	13
4.3.2 Modulvariablen.....	14

4.3.3 Temporäre Variablen.....	14
4.4 Module.....	14
4.5 Icons.....	14
5 Bezugsdokumente und -dateien.....	15
5.1 Bezugsdokumente.....	15
5.2 Bezugsdateien.....	15

Abkürzungs- und Akronymverzeichnis

ID	Identifikator
J	Jahr
KMU	kleine und mittlere Unternehmen
Min.	Minuten
MS	Microsoft
M	Monat
resp.	respectively
T	Tag
vgl.	vergleiche

Symbolverzeichnis

+/-	plus und minus
/	pro
%	Prozent

Tabellenverzeichnis

Tabelle 1: Übersicht Tabellenblätter Excel.....	3
Tabelle 2: Attribute in PACE.....	13

Abbildungsverzeichnis

Abbildung 1: Individuelle Icons.....	14
--------------------------------------	----

1 Problemstellung

Im Rahmen der vorliegenden Dissertation wird ein Baukastensystem für kleine und mittlere Unternehmen des Wirtschaftszweiges „Herstellung von Metallerzeugnissen“ erstellt. Zur Erstellung dieses Baukastensystems wurden zunächst das vorliegende PACE-Simulationsmodell `SIMULATIONSMODELL DISSERTATION.NET` (2016) sowie die vorliegende Excel-Datei `BEFÜLLUNG.XLSX` (2016) erstellt. Aus dem PACE-Simulationsmodell werden die Bausteine für das Baukastensystem sowie individuelle Icons für die Bausteine extrahiert. Die Excel-Datei wird unverändert in das Baukastensystem übernommen.

2 Rahmenbedingungen

2.1 Genutzte Software

Es wurden zwei Arten von Software verwendet. Zum einen wurde das Simulationswerkzeug PACE genutzt, im Folgenden kurz als PACE bezeichnet. Zum anderen wurde, zur Vereinfachung der Ein- und Ausgabe von Daten sowie um Fehleingaben durch den Anwender¹ zu vermeiden, die Software MS Excel verwendet, im Folgenden kurz als Excel bezeichnet.

PACE und Excel wurden über eine Schnittstelle miteinander gekoppelt, wobei der Datenaustausch von PACE aus gesteuert wird. Von PACE werden dabei sowohl Daten aus Excel eingelesen als auch Ergebnisse nach Excel zurückgeschrieben. Die Dateneingabe durch den Anwender erfolgt ausschließlich über Excel.

2.2 Einschränkungen

Aus technischen Gründen werden in Excel die folgenden Einschränkungen für die Dateneingabe vorgegeben:

- Es können maximal 100 Aufträge eingegeben werden.
- Es können maximal 50 Betriebs-Schließtage eingegeben werden.
- Es können maximal 50000 Werkstücke pro Los eingegeben werden.

¹ Es wird in diesem Dokument mehrfach der Begriff *Anwender* benutzt. Damit ist stets der Anwender des Baukastensystems (vgl. Kapitel 1, Seite 1) gemeint.

3 Excel

3.1 Allgemeines

Die Excel-Datei BEFÜLLUNG.XLSX (2016) besteht aus 10 Tabellenblättern. Die Funktion der einzelnen Tabellenblätter ist der folgenden Tabelle 2 zu entnehmen.

Tabellenblatt	Funktion	Erläuterung Funktion	Anmerkung
Anmerkungen	Hilfsblatt Anwender	Hier werden Informationen und Einschränkungen zur Dateneingabe aufgelistet.	geschützt
Umrechnung	Hilfsblatt Anwender	Dies ist ein Hilfsblatt zum Umrechnen von Bearbeitungszeiten in Dezimalzeiten sowie von Tagen in Minuten (dezi-mal).	teilweise geschützt
Betriebszeiten	Eingabe	Hier werden vom Anwender die Betriebszeiten sowie eventuelle Betriebsschließungen eingegeben.	teilweise geschützt
Aufträge	Eingabe	Hier werden vom Anwender die Daten der Aufträge, das Startdatum der Simulation und optional ein Abbruchdatum der Simulation eingegeben.	teilweise geschützt
Ergebnisse_ Aufträge	Ausgabe	Hier werden nach dem Simulationslauf die aufbereiteten Ergebnisse pro Auftrag aufgelistet. (Quelle: Tabellenblatt PACE_AUS)	geschützt
Ergebnisse_ Stationen	Ausgabe	Hier werden nach dem Simulationslauf die aufbereiteten Ergebnisse pro Bearbeitungsstation aufgelistet. (Quelle: Tabellenblatt PACE_AUS_Masch)	geschützt
PACE_EIN	Hilfsblatt PACE	Hier werden die Daten des Tabellenblattes Aufträge für PACE aufbereitet zur Verfügung gestellt. (PACE ließt hier aus)	ausgeblendet
PACE_AUS	Hilfsblatt PACE	Hier schreibt PACE während des Simulationslaufes die Ergebnisse pro Auftrag hinein.	ausgeblendet
PACE_AUS_ Masch	Hilfsblatt PACE	Hier schreibt PACE während des Simulationslaufes die Ergebnisse pro Bearbeitungsstation hinein.	ausgeblendet
Eingabewerte	Hilfsblatt Excel	Hier werden die Listenparameter für die Bearbeitungsstationen sowie für die Schwankungen festgelegt.	ausgeblendet

Tabelle 1: Übersicht Tabellenblätter Excel

Die Tabellenblätter sind in verschiedenen Farben gestaltet. Thematisch zusammengehörende Tabellenblätter sind in der gleichen Farbe gestaltet. Dies soll dem Anwender die Nutzung der Excel-Datei durch das Erfassen der Zusammenhänge erleichtern.

Des Weiteren sind die Tabellenblätter teilweise in farbige und weiße Bereiche unterteilt. Die Tabellenblätter sind so geschützt, dass nur die weißen Bereiche vom Anwender ausgefüllt werden können.

Der Schutz der Arbeitsmappe sowie der Tabellenblätter ist mit einem Passwort hinterlegt. Das Passwort lautet *Dissertation*.

Die Datei wurde mit einem Schreibschutz versehen.

3.2 Hilfsblätter Anwender

3.2.1 Anmerkungen

Im Tabellenblatt Anmerkungen sind Informationen und Einschränkungen aufgelistet, die dem Anwender die Nutzung der Excel-Datei erleichtern sollen.

3.2.2 Umrechnung

Im Tabellenblatt Umrechnung kann der Anwender Bearbeitungszeiten in Dezimalzeiten² sowie Tage in Minuten (dezimal) umrechnen.

3.3 Eingabe

3.3.1 Betriebszeiten

Im Tabellenblatt Betriebszeiten werden vom Anwender die Betriebszeiten sowie eventuelle Betriebsschließungen eingegeben.

In diesem Tabellenblatt werden einige Daten durch Excel automatisch erzeugt:

- Arbeitsminuten/Tag³

Es werden einige Daten aus anderen Tabellenblättern angezeigt:

- Start der Simulation⁴
- Abbruch⁵

Vom Anwender werden die folgenden allgemeinen Daten eingegeben:

- Arbeitszeit in Stunden/Tag⁶

Des Weiteren können maximal 50 Tage Betriebsschließung eingegeben werden:

2 Alle Dezimalzeiten in der Excel-Datei werden auf eine Nachkomma-Stelle gerundet.

3 Der Wert wird aus der eingegebenen Arbeitszeit in Feld E2 errechnet.

4 Der Wert wird aus Tabellenblatt Aufträge, Feld I1, übernommen.

5 Der Wert wird aus Tabellenblatt Aufträge, Feld O1, übernommen, sofern dort ein Wert eingetragen ist. Andernfalls wird der Wert „nein“ angezeigt.

6 **Datenüberprüfung:** Wenn der eingegebene Wert kleiner als 0,1 oder größer als 24 ist, wird eine Fehlermeldung „Ungültige Arbeitszeit“ angezeigt.

- Datum⁷
- Text (optional)

3.3.2 Aufträge

Im Tabellenblatt Aufträge werden vom Anwender die Daten der Aufträge eingegeben. Des Weiteren wird hier das Startdatum der Simulation eingegeben. Optional kann zusätzlich ein Datum für den Abbruch der Simulation eingegeben werden.

In diesem Tabellenblatt werden einige Daten durch Excel automatisch erzeugt:

- Aufträge gesamt⁸
- ID⁹

Vom Anwender werden die folgenden allgemeinen Daten eingegeben:

- Start der Simulation (morgens)¹⁰
- Abbruch der Simulation (abends, optional)¹¹

Des Weiteren werden die spezifischen Daten der Aufträge hier erfasst. Es wird pro Los¹² eine Zeile ausgefüllt, wobei die folgenden Daten eingegeben werden:

- Losgröße¹³
- Start¹⁴

7 **Datenüberprüfung:** Wenn der eingegebene Wert kein Datum des Formates TT.MM.JJ ist oder das Datum vor dem 01.01.00 liegt, wird eine Fehlermeldung „Ungültiges Datum“ angezeigt.

8 Der Wert wird aus den im Tabellenblatt Aufträge eingegebenen Aufträgen errechnet.

9 Der Wert wird pro Zeile eingetragen, sobald vom Anwender eine Losgröße eingegeben wird.

10 **Datenüberprüfung:** Wenn der eingegebene Wert kein Datum des Formates TT.MM.JJ ist oder das Datum vor dem 01.01.00 liegt, wird eine Fehlermeldung „Ungültiges Datum“ angezeigt.

Bedingte Formatierung: Das Feld wird rot hinterlegt, wenn kein Datum eingegeben wird oder das eingegebene Datum auf einen arbeitsfreien Tag fällt (Prüfung erfolgt über Wert in Tabellenblatt PACE_EIN Feld G1).

11 **Datenüberprüfung:** Wenn der eingegebene Wert kein Datum des Formates TT.MM.JJ ist oder das Datum vor dem Datum „Start der Simulation“ liegt, wird eine Fehlermeldung „Ungültiges Datum“ angezeigt.

Bedingte Formatierung: Das Feld wird rot hinterlegt, wenn das eingegebene Datum auf einen arbeitsfreien Tag fällt (Prüfung erfolgt über Wert in Tabellenblatt PACE_EIN Feld J1).

12 Die Bezeichnungen *Aufträge* und *Lose* werden in Excel synonym verwendet.

Der Begriff *Auftrag* wird als allgemeine Bezeichnung verwendet („ein Kunde gibt einen *Auftrag*“), sobald der *Auftrag* jedoch in der Produktion gegeben wird oder für die Produktion genauer spezifiziert wird, wird aufgrund der Fachsprache der Anwender der Begriff *Los* verwendet. („ein *Los* wird bearbeitet“).

13 **Datenüberprüfung:** Wenn der eingegebene Wert kleiner als 1 oder größer als 50000, ist wird eine Fehlermeldung „Ungültige Losgröße“ angezeigt.

14 **Datenüberprüfung:** Wenn der eingegebene Wert kein Datum des Formates TT.MM.JJ ist oder das Datum vor dem 01.01.00 liegt, wird eine Fehlermeldung „Ungültiges Datum“ angezeigt.

Bedingte Formatierung: Das Feld wird rot hinterlegt wenn kein Datum eingegeben wird oder das eingegebene Datum auf einen arbeitsfreien Tag fällt (Prüfung erfolgt über Wert in Tabellenblatt PACE_EIN Spalte BB).

Bedingte Formatierung: Das Feld wird gelb hinterlegt, wenn das eingegebene Datum vor dem Simulationsstart liegt (Prüfung erfolgt über Wert in Tabellenblatt Aufträge Feld I1).

- Fertig soll (optional)¹⁵

Es können pro Los bis zu 10 Bearbeitungsstationen erfasst werden:

- Station (Pull-down-Auswahlmenü)¹⁶
- Rüstzeit/Los (Min.)¹⁷
- Schwankung Rüstzeit, jeweils +/- (%) (Pull-down-Auswahlmenü)¹⁸
- Bearbeitung/Stück (Min.)¹⁹
- Schwankung Bearbeitung, jeweils +/- (%) (Pull-down-Auswahlmenü)²⁰

3.4 Ausgabe

3.4.1 Ergebnisse_Aufträge

Im Tabellenblatt Ergebnisse_Aufträge werden nach dem Simulationslauf die aufbereiteten Ergebnisse (basierend auf dem Tabellenblatt PACE_AUS) für die Aufträge ausgegeben. Es wird pro Auftrag eine Zeile aufgelistet. Die folgenden Daten werden dargestellt:

- ID²¹
- Start Auftrag SOLL²²
- Start Auftrag IST²³
- Fertigstellung SOLL²⁴
- Fertigstellung IST²⁵
- Durchlaufzeit (Min.)²⁶

15 **Datenüberprüfung:** Wenn der eingegebene Wert kein Datum des Formates TT.MM.JJ ist oder das Datum vor dem Datum „Start Auftrag“ liegt, wird eine Fehlermeldung „Ungültiges Datum“ angezeigt.

Bedingte Formatierung: Das Feld wird rot hinterlegt, wenn das eingegebene Datum auf einen arbeitsfreien Tag fällt (Prüfung erfolgt über Wert in Tabellenblatt PACE_EIN Spalte BC).

16 Die Listenparameter sind in dem Tabellenblatt Eingabewerte festgelegt.

17 **Datenüberprüfung:** Wenn der eingegebene Wert kleiner als 0 ist, wird eine Fehlermeldung „Ungültige Rüstzeit“ angezeigt.

18 Die Listenparameter sind in dem Tabellenblatt Eingabewerte festgelegt. PACE nutzt eine Dreiecksverteilung.

19 **Datenüberprüfung:** Wenn der eingegebene Wert kleiner als 0 ist, wird eine Fehlermeldung „Ungültige Bearbeitungszeit“ angezeigt.

20 Die Listenparameter sind in dem Tabellenblatt Eingabewerte festgelegt. PACE nutzt eine Dreiecksverteilung.

21 Der Wert wird aus dem Tabellenblatt PACE_AUS, Spalte A, übernommen.

22 Der Wert wird aus dem Tabellenblatt Aufträge, Spalte C, übernommen.

23 Der Wert wird aus dem Tabellenblatt Ergebnisse_Aufträge, Spalte U, übernommen.

24 Der Wert wird aus dem Tabellenblatt Aufträge, Spalte D, übernommen.

25 Der Wert wird aus dem dem Tabellenblatt PACE_AUS, Spalte F, mittels einer Funktion errechnet. Sollte der Wert auf einen arbeitsfreien Tag fallen, wird der darauf folgende Arbeitstag angezeigt.

Bedingte Formatierung: Das Datum wird fett rot angezeigt, wenn es nach dem Datum „Fertigstellung SOLL“ (Tabelleblatt Ergebnisse_Aufträge, Spalte D) liegt. Andernfalls wird das Datum fett grün angezeigt.

26 Der Wert wird aus dem Tabellenblatt PACE_AUS, Spalte G, übernommen und auf eine Nachkommastelle gerundet.

- Rüstzeit
 - Zeit (Min.)²⁷
 - in %²⁸
- Bearbeitungszeit
 - Zeit (Min.)²⁹
 - in %³⁰
- Wartezeit
 - Zeit (Min.)³¹
 - in %³²

Für die Produktionsplanung werden pro Los bis zu 10 Bearbeitungsstationen aufgelistet:

- Produktionsplanung
 - Station³³
 - Start auf Station³⁴

3.4.2 Ergebnisse_Stationen

Im Tabellenblatt Ergebnisse_Stationen werden nach dem Simulationslauf die aufbereiteten Ergebnisse (basierend auf dem Tabellenblatt PACE_AUS_Masch) für die Bearbeitungsstationen ausgegeben. Die folgenden allgemeinen Daten werden dargestellt:

- Simulierte Zeit in Minuten³⁵

Es wird für jede Kategorie von Bearbeitungsstationen eine Zeile aufgelistet. Die folgenden Daten werden aufgelistet:

- Kategorie Bearbeitungsstation³⁶

27 Der Wert wird aus dem Tabellenblatt PACE_AUS, Spalte D, übernommen und auf eine Nachkommastelle gerundet.

28 Der Wert aus den Werten der Spalten H und F des Tabellenblattes Ergebnisse_Aufträge mittels einer Funktion errechnet.

29 Der Wert wird aus dem Tabellenblatt PACE_AUS, Spalte E, übernommen und auf eine Nachkommastelle gerundet.

30 Der Wert aus den Werten der Spalten L und F des Tabellenblattes Ergebnisse_Aufträge mittels einer Funktion errechnet.

31 Der Wert wird aus dem Tabellenblatt PACE_AUS, Spalte C, übernommen und auf eine Nachkommastelle gerundet.

32 Der Wert aus den Werten der Spalten P und F des Tabellenblattes Ergebnisse_Aufträge mittels einer Funktion errechnet.

33 Die Werte werden aus dem Tabellenblatt PACE_AUS, Spalten H, J, L, N, P, R, T, V, X, Z, übernommen.

34 Die Werte werden aus dem Tabellenblatt PACE_AUS, Spalten I, K, M, O, Q, S, U, W, Y, AA, mittels einer Funktion errechnet. Sollte der Wert auf einen arbeitsfreien Tag fallen, wird der darauf folgende Arbeitstag angezeigt.

35 Der Wert wird aus dem Tabellenblatt PACE_AUS_Masch, Feld G1, übernommen.

36 Der Wert wird aus dem Tabellenblatt PACE_AUS_Masch, Spalte A, übernommen.

- Betriebszeit Stationen³⁷
- Rüstzeit
 - Zeit (Min.)³⁸
 - in %³⁹
- Bearbeitungszeit⁴⁰
 - Zeit (Min.)⁴¹
 - in %⁴²
- Wartezeit
 - Zeit (Min.)⁴³
 - in %⁴⁴
- Anzahl⁴⁵

3.5 Hilfsblätter PACE

3.5.1 PACE_EIN

PACE greift auf dieses Tabellenblatt PACE_EIN zu und liest Daten aus. Daher werden in diesem Tabellenblatt die Daten für PACE aufbereitet zur Verfügung gestellt.

Das Tabellenblatt ist ausgeblendet und somit für den Anwender nicht sichtbar.

Es werden die folgenden Daten dargestellt:

- Aufträge gesamt⁴⁶
- Start⁴⁷

³⁷ Der Wert wird aus dem Tabellenblatt Ergebnisse_Stationen, Spalten D und H, mittels einer Funktion errechnet.
³⁸ Der Wert wird aus dem Tabellenblatt PACE_AUS_Masch, Spalte B, übernommen und auf eine Nachkommastelle gerundet.
³⁹ Der Wert wird aus dem Tabellenblatt Ergebnisse_Stationen, Feld C1 sowie Spalten D und T, mittels einer Funktion errechnet und auf eine Nachkommastelle aufgerundet.
⁴⁰ Bei den Kategorien Bearbeitungsstationen *gleitschleif* und *extern* wird hier keine Zeit aufgegeben.
⁴¹ Der Wert wird aus dem Tabellenblatt PACE_AUS_Masch, Spalte C, übernommen. und auf eine Nachkommastelle gerundet.
⁴² Der Wert wird aus dem Tabellenblatt Ergebnisse_Stationen, Feld C1 sowie Spalten H und T, mittels einer Funktion errechnet und auf eine Nachkommastelle aufgerundet.
⁴³ Der Wert wird aus dem Tabellenblatt Ergebnisse_Stationen, Feld C1 sowie Spalten B und T, mittels einer Funktion errechnet.
⁴⁴ Der Wert wird aus dem Tabellenblatt Ergebnisse_Stationen, Feld C1 sowie Spalten P und T, mittels einer Funktion errechnet und auf eine Nachkommastelle aufgerundet.
⁴⁵ Der Wert wird aus dem Tabellenblatt PACE_AUS_Masch, Spalte E, übernommen.
⁴⁶ Der Wert wird aus dem Tabellenblatt Aufträge, Feld C1, übernommen.
⁴⁷ Der Wert wird aus dem Tabellenblatt Aufträge, Feld I1, übernommen.

- Feld G1⁴⁸
- Abbruch⁴⁹
- Feld J1⁵⁰

Des Weiteren werden die spezifischen Daten der Aufträge dargestellt. Es wird pro Los eine Zeile mit folgenden Daten dargestellt:

- Los ID⁵¹
- Losgröße⁵²
- Start Auftrag⁵³

Es werden pro Los bis zu 10 Bearbeitungsstationen aufgelistet:

- Station⁵⁴
- Rüstzeit/Los (Min.)⁵⁵
- Schwankung Rüstzeit, jeweils +/- (%)⁵⁶
- Bearbeitung/Stück (Min.)⁵⁷
- Schwankung Bearbeitung, jeweils +/- (%)⁵⁸

Rechts neben den Bearbeitungsstationen⁵⁹ werden noch zwei weitere Daten dargestellt:

- Abfrage freier Starttag⁶⁰
- Abfrage freier Fertigtag⁶¹

48 Der Wert wird aus dem Tabellenblatt Aufträge, Felder I1, sowie dem Tabellenblatt Betriebszeiten, Felder A5-A55, errechnet.

49 Der Wert wird aus dem Tabellenblatt Aufträge, Felder I1 und O1, sowie dem Tabellenblatt Betriebszeiten, Felder A5-A55 und F1, errechnet.

50 Der Wert wird aus dem Tabellenblatt Aufträge, Feld O1, sowie dem Tabellenblatt Betriebszeiten, Felder A5-A55 und H2, errechnet.

51 Der Wert wird aus dem Tabellenblatt Aufträge, Spalte A, übernommen.

52 Der Wert wird aus dem Tabellenblatt Aufträge, Spalte B, übernommen.

53 Der Wert wird aus dem Tabellenblatt PACE_EIN, Spalte B, dem Tabellenblatt Aufträge, Spalte C und Feld I1, sowie dem Tabellenblatt Betriebszeiten, Felder A5-A55 und Feld H2, errechnet.

54 Die Werte werden aus dem Tabellenblatt Aufträge, Spalten E, N, W, AF, AO, AX, BG, BP, BY, CH, sowie dem Tabellenblatt Eingabewerte, Felder A3-B13, errechnet.

55 Die Werte werden aus dem Tabellenblatt Aufträge, Spalten F, O, X, AG, AP, AY, BH, BQ, BZ und CI, übernommen und auf eine Nachkommastelle gerundet.

56 Die Werte werden aus dem Tabellenblatt Aufträge, Spalten H, Q, Z, AI, AR, BA, BJ, BS, CB, und CK, übernommen.

57 Die Werte werden aus dem Tabellenblatt Aufträge, Spalten J, S, AB, AK, AT, BC, BL, BU, CD und CM, übernommen und auf eine Nachkommastelle gerundet.

58 Die Werte werden aus dem Tabellenblatt Aufträge, Spalten L, U, AD, AM, AV, BE, BN, BW, CF und CO, übernommen.

59 In den Spalten BB und BC.

60 Der Wert wird aus dem Tabellenblatt PACE_EIN, Spalte A, dem Tabellenblatt Aufträge, Spalte C, sowie dem Tabellenblatt Betriebszeiten, Felder A5-A55, errechnet.

61 Der Wert wird aus dem Tabellenblatt PACE_EIN, Spalte A, dem Tabellenblatt Aufträge, Spalte D, sowie dem Tabellenblatt Betriebszeiten, Felder A5-A55, errechnet.

3.5.2 PACE_AUS

PACE greift auf dieses Tabellenblatt PACE_AUS zu und trägt während des Simulationslaufes die Ergebnisse der simulierten Aufträge ein.

Das Tabellenblatt ist ausgeblendet und somit für den Anwender nicht sichtbar.

Es wird pro Los eine Zeile mit folgenden Daten dargestellt:

- ID
- Start Los
- Warte
- Rüst
- Bearb
- Ende
- DLZ

Es werden pro Los bis zu 10 Bearbeitungsstationen aufgelistet:

- Station
- Start auf Station

3.5.3 PACE_AUS_Masch

PACE greift auf dieses Tabellenblatt PACE_AUS_Masch zu und trägt während des Simulationslaufes die Ergebnisse der Bearbeitungsstationen ein.

Das Tabellenblatt ist ausgeblendet und somit für den Anwender nicht sichtbar.

Es werden die folgenden Daten dargestellt:

- Betriebszeit

Es wird pro Kategorie Bearbeitungsstation eine Zeile mit folgenden Daten aufgelistet:

- Station
- rüsten
- bearbeiten
- warten

- Anzahl

3.6 Hilfsblatt Excel: Eingabewerte

Im Tabellenblatt Eingabewerte werden die Listenparameter festgelegt.

Das Tabellenblatt ist ausgeblendet und somit für den Anwender nicht sichtbar.

Es werden die folgenden Daten dargestellt:

- verfügbare Bearbeitungsstationen
- Schwankung (%)

4 PACE

4.1 Dokumentation

Das SIMULATIONSMODELL DISSERTATION.NET (2016) wurde in PACE mittels mehrerer Arten dokumentiert. Die komplette Dokumentation in PACE ist dem Dokument RUDEL (2016) zu entnehmen⁶².

Für jedes Modul wurde eine *Development Info*⁶³ hinterlegt. Dort wurden jeweils folgende Informationen eingetragen:

- Modul
- Entwickler
- Version
- Datum

Ebenfalls wurde für jedes Modul eine Purpose Description angelegt. Dort wurden jeweils folgende Informationen eingetragen:

- Modulname
- Aufgabe
- Eingangsstellen
- Ausgangsstellen
- eventuelle Modulvariablen

In den *Action Codes*, welche an den Transitionen des Petrinetz-Graphen ausgeführt werden, wurde der Zweck des jeweiligen Codefragments „auskommentiert“ umgangssprachlich erläutert.

4.2 Attribute

In PACE werden Attribute verwendet, die beim Ausführen des Markenspiels mit den Marken durch den Petrinetz-Graphen wandern.

Die Attribute werden in der folgenden Tabelle 2 aufgelistet und erläutert.

⁶² Vgl. den Anhang A6 der vorliegenden Dissertation.

⁶³ Im Simulationswerkzeug PACE lautet die Bezeichnung *Development Info*, in der Dokumentation RUDEL (2016) lautet die Bezeichnung *Development Comment* (beide Bezeichnungen sind von PACE systemtechnisch vergeben).

Attribut	Bezeichnung	Art	Beispiel	Erläuterung
idL	Los-ID	natürliche Zahl ohne Null	1	eindeutige Nummer des Loses, in Excel festgelegt
lg	Losgröße	natürliche Zahl ohne Null	100	Anzahl der Werkstücke in dem Los, in Excel festgelegt
status	Status des Auftrages	String	#wartet	zeigt den aktuellen Status des Loses während der Simulation an; kann die Werte #eingelassen, #wartet, #rüstet, #bearbeitung und #fertig annehmen
start	Startzeit des Auftrages	Dezimalzeit	255	gibt die Simulationszeit an, zu der das Los in die Produktion gegeben wird
bST	Bearbeitungsstation	Symbol	#cncDreh	gibt an, auf welcher Bearbeitungsstation das Los als nächstes bearbeitet wird
tR	Rüstzeit/Los	Dezimalzeit	20,5	gibt die Rüstzeit des Loses für die Bearbeitungsstation an
tB	Bearbeitungszeit/Werkstück	Dezimalzeit	2,3	gibt die Bearbeitungszeit je Werkstück dieses Loses für die Bearbeitungsstation an
cEIN	Container EIN	Ordered Collection	(#cncDreh 20 2)	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten speichert
cAUS	Container AUS	Dictionary	#cncFräs (10 0)	Dictionary, welche die Auswerte-Daten je Los speichert
zähler	Zähler	natürliche Zahl	18	wird zum Einlesen der Aufträge aus Excel benötigt
aufträge	Anzahl der Aufträge	natürliche Zahl	5	Anzahl der in Excel eingegebenen Aufträge
tEIN	Eingangszeit Zwischenlager	Dezimalzeit	18,9	wird zur Auswertung der Wartezeit im Zwischenlager benötigt
t1	Zeit 1	Dezimalzeit	50,2	wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt
t2	Zeit 2	Dezimalzeit	266,9	wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt

Tabelle 2: Attribute in PACE

4.3 Variablen

4.3.1 Globale Variablen

In dem SIMULATIONSMODELL DISSERTATION.NET (2016) werden globale Variablen verwendet. Diese können dem Dokument RUDEL (2016) unter dem Menüpunkt *Global Variables* entnommen werden.

4.3.2 Modulvariablen

In dem SIMULATIONSMODELL DISSERTATION.NET (2016) werden bei einigen Modulen Modulvariablen verwendet. Diese können dem Dokument RUDEL (2016) in den einzelnen Modulen unter dem jeweiligen Menüpunkt *Net resp. Module Variables* entnommen werden.

4.3.3 Temporäre Variablen

Neben den Attributen werden im Code der Transitionen⁶⁴ des Petrinetz-Graphen temporäre Variablen verwendet. Diese temporären Variablen können dem Dokument RUDEL (2016) unter den jeweiligen Menüpunkten *Condition Code*, *Delay Code* und *Action Code* entnommen werden.

4.4 Module

Die Module können dem Dokument RUDEL (2016) entnommen werden. Die Beschreibung der Module ist unter dem jeweiligen Menüpunkt *Purpose Description* erläutert.

4.5 Icons

Im Rahmen der Erstellung des SIMULATIONSMODELL DISSERTATION.NET (2016) wurden individuelle Icons für die Bausteine erstellt. Die Icons sind in der Datei SIMULATIONSMODELL DISSERTATION.ICN (2016) hinterlegt.

Die folgende Abbildung 1 zeigt die individuellen Icons.

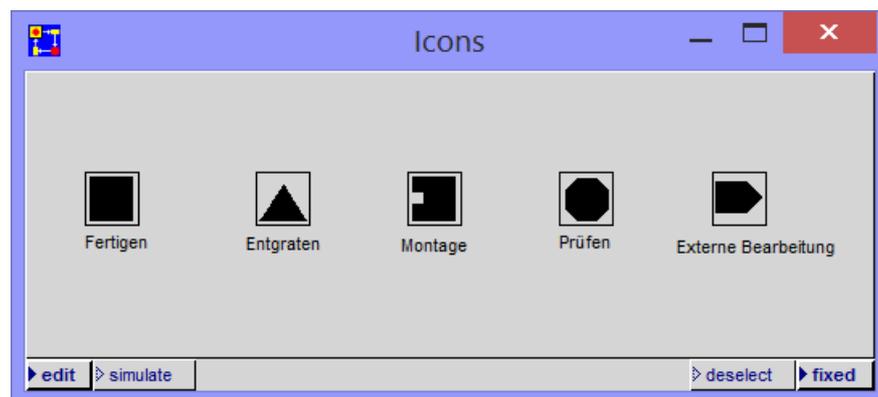


Abbildung 1: Individuelle Icons

64 *Condition Code, Delay Code, Action Code.*

5 Bezugsdokumente und -dateien

5.1 Bezugsdokumente

RUDEL (2016)

RUDEL, S.: PACE model: Simulationsmodell Dissertation – January 20, 2016 2:55:12 pm.
Version 1.0. Unterhaching 2016.

5.2 Bezugsdateien

BEFÜLLUNG.XLSX (2016)

RUDEL, S.: Befüllung.xlsx. MS Excel-Datei, Version 1.0. Unterhaching 2016.

SIMULATIONSMODELL DISSERTATION.ICN (2016)

RUDEL, S.: Simulationsmodell Dissertation.icn. PACE Icon-Datei, Version 1.0. Unterhaching 2016.

SIMULATIONSMODELL DISSERTATION.NET (2016)

RUDEL, S.: Simulationsmodell Dissertation.net. PACE Netz-Datei, Version 1.0. Unterhaching 2016.

Anhang A6

Simulationsmodell Dissertation
(Automatische Dokumentation aus PACE)

Global Variables:

ExcelMaschAus
Kanal1
Kanal2
Kanal3

PACE model: Simulationsmodell Dissertation

January 20, 2016 2:55:12 pm

Modules

Simulationsmodell Dissertation

 Maschinenpark

 Kategorie Bearbeitungsstation

 Bearbeitungsstation

 Station rüsten

 Werkstücke bearbeiten

 Control_Bearbeitungsstation

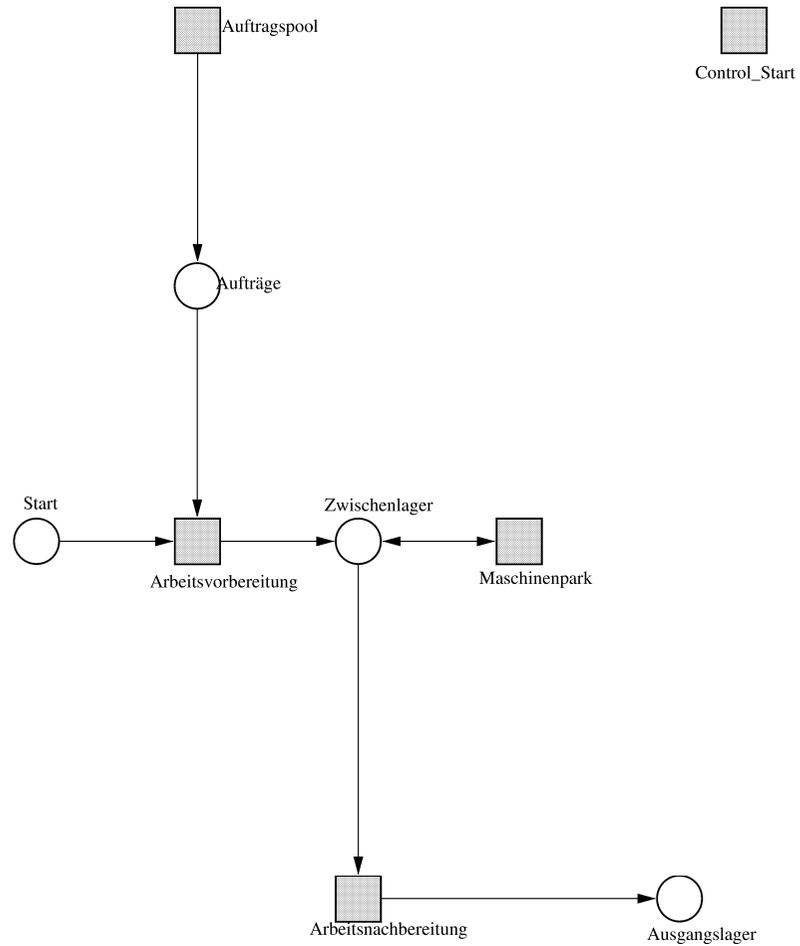
 Control_Start

 Auftragspool

 Arbeitsvorbereitung

 Arbeitsnachbereitung

Module: Simulationsmodell Dissertation



Module: Maschinenpark



====> Development Comment <====

Modul: Maschinenpark
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Maschinenpark

Aufgabe: Dieses Modul "Maschinenpark" beinhaltet die Bearbeitungsstationen.

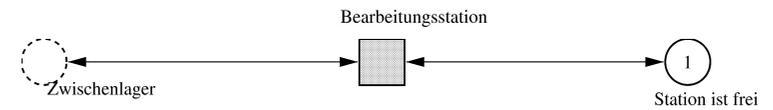
Anzahl der Eingangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Modul "Bearbeitungsstation".

Anzahl der Ausgangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Modul "Bearbeitungsstation".

Module: Kategorie Bearbeitungsstation



====> Development Comment <====

Modul: Kategorie Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Kategorie Bearbeitungsstation

Aufgabe: Dieses Modul "Kategorie Bearbeitungsstation" beinhaltet alle Bearbeitungsstationen EINES Typs (z.B. alle CNC-Drehmaschinen). Der Typ der Bearbeitungsstationen wird über die Modulvariable #maschine definiert. Die Anzahl der Bearbeitungsstationen in diesem Modul wird über die eingefügte Anzahl der Module "Bearbeitungsstation" eine Ebene tiefer definiert.

Anzahl der Eingangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Anzahl der Ausgangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Modulvariable Vorbesetzung (Initial) Beschreibung

#anzahl 0 Anzahl der Bearbeitungsstationen in diesem Modul.
#bearbeiten 0 Summiert die Bearbeitungszeiten der Bearbeitungsstationen während der Simulation auf.
#rüsten 0 Summiert die Rüstzeiten der Bearbeitungsstationen während der Simulation auf.
#station #eintragen Definition des Typs der Bearbeitungsstationen.
#warten 0 Summiert die Wartezeiten der Bearbeitungsstationen während der Simulation auf.

====> Net resp. Module Variables <====

#anzahl
#bearbeiten
#rüsten
#station

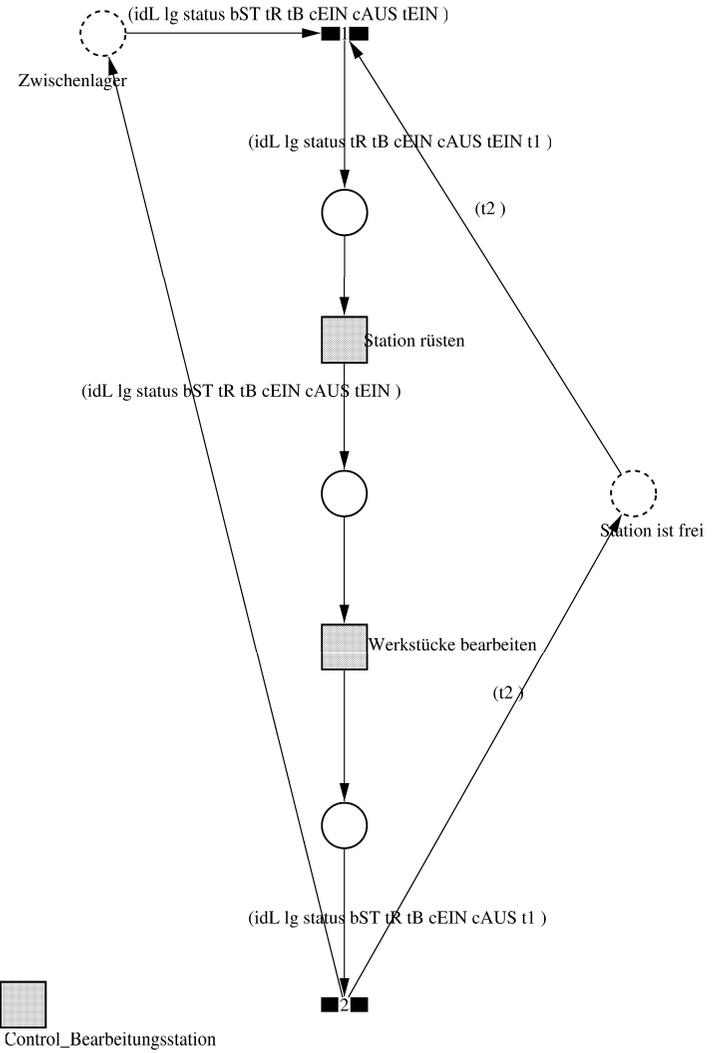
#warten

1: Place 'Station ist frei'

====> Initial Tokens <====

(CurrentTime)

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation
 Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2
 Bezeichner Attribute Beschreibung

- | Bezeichner | Attribute | Beschreibung |
|----------------------|--|--------------|
| Zwischenlager idL | Eindeutige Nummer des Loses, in Excel festgelegt. | |
| Zwischenlager lg | Anzahl der Werkstücke in dem Los, in Excel festgelegt. | |
| Zwischenlager status | Zeigt den aktuellen Status des Loses in der Simulation an. | |
| Zwischenlager bST | Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird. | |
| Zwischenlager tR | Gibt die Rüstzeit des Loses für die Bearbeitungsstation an. | |
| Zwischenlager tB | Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an. | |
| Zwischenlager cEIN | Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert. | |
| Zwischenlager cAUS | Collection, welche die Auswerte-Daten des Loses speichert. | |
| Zwischenlager tEIN | Wird zur Auswertung der Wartezeit im Zwischenlager benötigt. | |
| Station ist frei t2 | Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt. | |

Anzahl der Ausgangsstellen: 2
 Bezeichner Attribute Beschreibung

- | Bezeichner | Attribute | Beschreibung |
|----------------------|--|--------------|
| Zwischenlager idL | Eindeutige Nummer des Loses, in Excel festgelegt. | |
| Zwischenlager lg | Anzahl der Werkstücke in dem Los, in Excel festgelegt. | |
| Zwischenlager status | Zeigt den aktuellen Status des Loses in der Simulation an. | |
| Zwischenlager bST | Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird. | |
| Zwischenlager tR | Gibt die Rüstzeit des Loses für die Bearbeitungsstation an. | |
| Zwischenlager tB | Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an. | |
| Zwischenlager cEIN | Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert. | |
| Zwischenlager cAUS | Collection, welche die Auswerte-Daten des Loses speichert. | |

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel

====> Action Code <====

t2:= CurrentTime.
tEIN:= (CurrentTime) value.

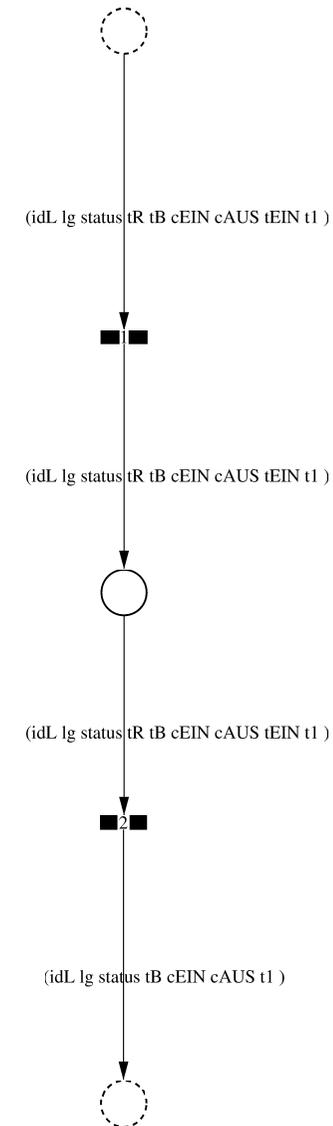
"Status Los umsetzen"
status:= #wartet.

"Status der Bearbeitungsstation in Excel schreiben"
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:= (((rüsten*10) rounded)//10)
printString, ', ', (((rüsten*10) rounded)\10)*10) printString.
bearbeitenExcel:= (((bearbeiten*10) rounded)//10)
printString, ', ', (((bearbeiten*10) rounded)\10)*10) printString.
wartenExcel:= (((warten*10) rounded)//10)
printString, ', ', (((warten*10) rounded)\10)*10) printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].  
(station =#manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].  
(station =#manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].  
(station =#montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"  
betriebszeit:= CurrentTime value.  
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)  
printString, ', ',(((betriebszeit*10)rounded)\10)*10)printString.  
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"

tRNetto:= tR removeFirst.

tRVarProzent:= tR removeFirst.

(tRVarProzent > 0) ifTrue:

[tRVar:= ((tRVarProzent/100) *tRNetto).

tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]

ifFalse:

[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"

tRüstAlt:= cAUS at: #tRüst.

tRüstHier:= ((CurrentTime) value) - tEIN.

tRüstNeu:= tRüstAlt + tRüstHier.

cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"

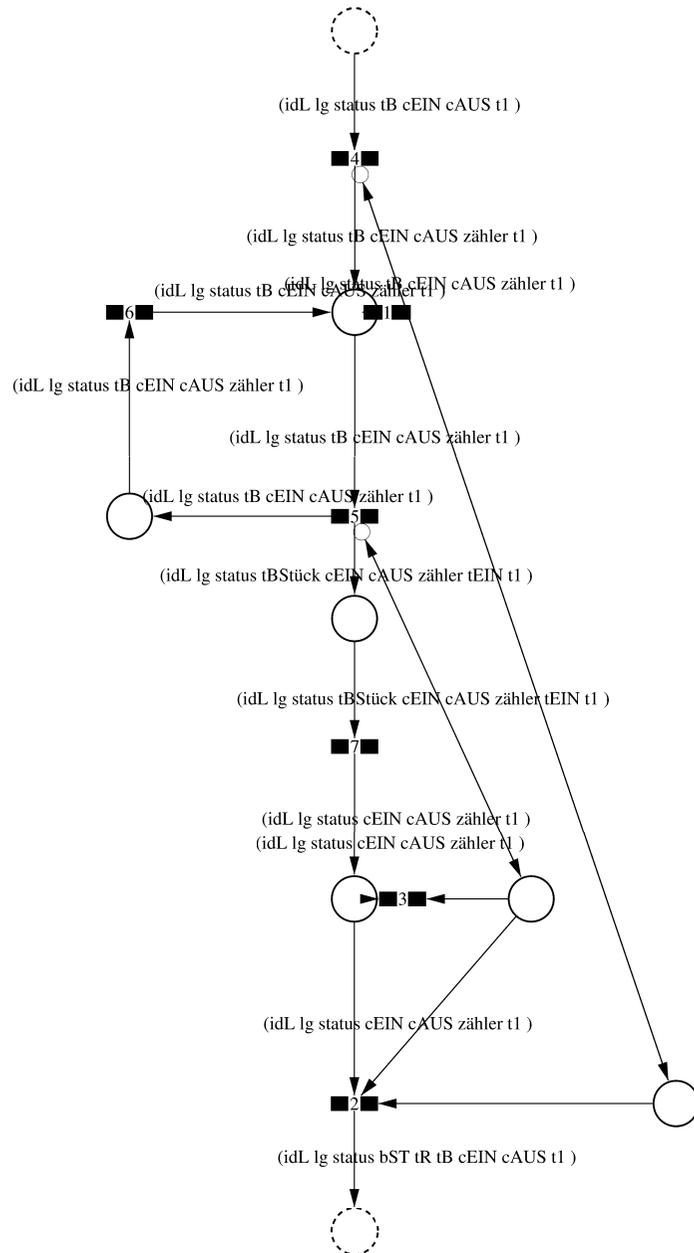
t2:= (CurrentTime) value.

rüstenAlt:= (self at: #rüsten) value.

rüstenNeu:= (t2-tEIN) +rüstenAlt.

(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- | Bezeichner | Attribute | Beschreibung |
|------------|--|--------------|
| -- idL | Eindeutige Nummer des Loses, in Excel festgelegt. | |
| -- lg | Anzahl der Werkstücke in dem Los, in Excel festgelegt. | |
| -- status | Zeigt den aktuellen Status des Loses in der Simulation an. | |
| -- tB | Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an. | |
| -- cEIN | Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert. | |
| -- cAUS | Collection, welche die Auswerte-Daten des Loses speichert. | |
| -- t1 | Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt. | |

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- | Bezeichner | Attribute | Beschreibung |
|------------|--|--------------|
| -- idL | Eindeutige Nummer des Loses, in Excel festgelegt. | |
| -- lg | Anzahl der Werkstücke in dem Los, in Excel festgelegt. | |
| -- status | Zeigt den aktuellen Status des Loses in der Simulation an. | |
| -- bST | Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird. | |
| -- tR | Gibt die Rüstzeit des Loses für die Bearbeitungsstation an. | |
| -- tB | Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an. | |
| -- cEIN | Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert. | |
| -- cAUS | Collection, welche die Auswerte-Daten des Loses speichert. | |
| -- t1 | Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt. | |

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
if True:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
if False:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"
zähler:= 1

5: Transition

====> Transition Variables <====

#zähler
#tBNetto
#tBVar
#tBStück
#tBVarProzent
#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"
tBNetto:= tB at: 1.
tBVarProzent:= tB at: 2.
(tBVarProzent > 0) if True:
 [tBVar:= ((tBVarProzent/100) *tBNetto).
 tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]
if False:
 [tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"
status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"
zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt
#tBearbHier
#tBearbNeu
#t2
#bearbeitenAlt
#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"
tBearbAlt:= cAUS at: #tBearb.
tBearbHier:= ((CurrentTime) value) - tEIN.
tBearbNeu:= tBearbAlt + tBearbHier.
cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
bearbeitenAlt:= (self at: #bearbeiten) value.
bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.
(self at: #bearbeiten) value: arbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.

Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

#anzahlAlt
#anzahlNeu
#station

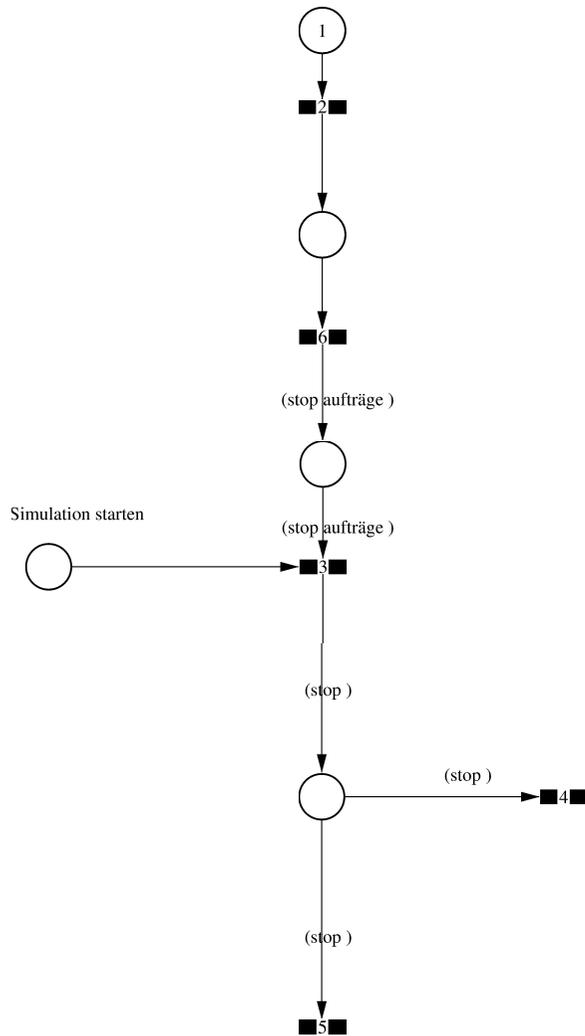
====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"

anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station = #manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Control_Start



====> Development Comment <====

Modul: Control_Start
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Start

Aufgabe: Dieses Modul "Control_Start" prüft ob die Excel-Datei "Befüllung.xlsx" geöffnet ist, öffnet die Kanäle dorthin und überschreibt alte Ergebnisse in Excel mit "0".

Es wird die Anzahl der Aufträge sowie ein evtl. eingegebener vorzeitiger Stop der Simulation aus Excel eingelesen.

Anschließend werden so viele Token wie Aufträge in Excel eingegeben wurden in die Stelle "Start Aufträge" im Modul "Auftragspool" eingelegt.

Sobald die Aufträge im Modul "Auftragspool" eingelesen sind, wird von dort ein Token zurück in die Stelle "Simulation starten" dieses Moduls "Control_Start" gelegt.

Anschließend werden so viele Token in die Stelle "Start" gelegt wie Aufträge in Excel eingegeben wurden.

Im weiteren Verlauf ist dieses Modul nur noch relevant, falls ein vorzeitiger Stop der Simulation in Excel eingetragen wurde. In diesem Fall stoppt die Simulation nach der angegebenen Zeit.

Dieses Modul "Control_Start" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

#abfrage
#testvariable

#aufträge
#excelAus
#container

====> Action Code <====

"Prüfen ob die Excel-Datei Befüllung.xlsx geöffnet ist"
abfrage:= DialogView confirm: 'Beim nun folgenden Start der Simulation kann es zu einer Wartezeit von bis zu einer Minute kommen, in der Sie keine Reaktion des Simulationswerkzeuges PACE sehen können. Bitte haben Sie Geduld, in dieser Zeit werden die Aufträge aus der Excel-Datei Befüllung.xlsx eingelesen.

Ist die Excel-Datei Befüllung.xlsx geöffnet?

,
initialAnswer: true.
(abfrage = false) ifTrue: [DialogView warn:'Bitte öffnen Sie zunächst die Excel-Datei und starten Sie den Lauf des PACE-Modells anschließend neu.'. self terminate.].
(abfrage = true) ifTrue: [
"Test ob wirklich offen"
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.
aufträge:= (DDE getData: Kanal3 item:'Z1S3') asNumber].

"Vorherige Ergebnisse aus Excel löschen, Blatt PACE_AUS_Masch; das MUSS hier geschehen, da ein Initial Token im Modul Control_Bearbeitungsstation direkt als nächstes läuft und die Anzahl der Bearbeitungsstationen einträgt!"
excelAus:= OrderedCollection new.
11 timesRepeat: [
container:= OrderedCollection new.
4 timesRepeat:[container addFirst: 0.].
excelAus addFirst: container.].
DDE writeNumbersToExcel: Kanal3 startAt: 'Z2S2' dataList: excelAus.
DDE putData: Kanal3 item: 'Z1S7' data: 0 printString.

3: Transition

====> Transition Variables <====

#startSimulation

====> Action Code <====

"Simulation starten"
startSimulation:= self placeNamed: 'Start'.
aufträge timesRepeat: [self addTokenTo: startSimulation].

4: Transition

====> Transition Variables <====

#stop

====> Condition Code <====

stop = 0

5: Transition

====> Condition Code <====

stop > 0

====> Delay Code <====

stop

====> Action Code <====

"Vorzeitigen Stop Simulation ausführen"
DialogView warn: 'Die Simulation wurde zu dem von Ihnen in Excel eingegeben Datum beendet.
Die Daten der Bearbeitungsstationen sind im Tabellenblatt Ergebnisse_Stationen in Excel ablesbar.

Möglicherweise wurden nicht alle Aufträge fertiggestellt - nicht fertiggestellte Aufträge werden
im Tabellenblatt Ergebnisse_Aufträge in Excel NICHT angezeigt.
Die Status dieser nicht fertiggestellten Aufträge können jedoch in PACE abgelesen werden.'

"Aktuelle Zeit als Ende Betriebszeit in Excel schreiben"
DDE putData: Kanal3 item: 'Z1S7' data: stop printString.
self terminate.

6: Transition

====> Transition Variables <====

#startAufträge
#abfrage
#aufträge

```
#excelAus  
#container  
#stop  
#startFrei  
#stoppFrei
```

====> Action Code <====

"Kanäle öffnen und Auftragsanzahl einlesen"

```
Kanal1:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS'.  
Kanal2:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_EIN'.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
aufträge:= (DDE getData: Kanal2 item:'Z1S3') asNumber.
```

"Vorherige Ergebnisse aus Excel löschen, Blatt PACE_AUS"

```
excelAus:= OrderedCollection new.  
100 timesRepeat: [  
  container:= OrderedCollection new.  
  27 timesRepeat:[container addFirst: 0].  
  excelAus addFirst: container.].  
DDE writeNumbersToExcel: Kanal1 startAt: 'Z4S1' dataList: excelAus.
```

"Prüfen ob Simulationsstart arbeitsfreier Tag"

```
startFrei:= (DDE getData: Kanal2 item:'Z1S7') asNumber.  
(startFrei = 99) ifTrue:[DialogView warn: '  
  Sie haben in der Excel-Tabelle für die Simulation keine Startzeit eingegeben.  
  Das Feld ist in Excel im Tabellenblatt Aufträge (oben) rot hinterlegt.  
  Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE  
  anschließend neu.'. self terminate.].  
(startFrei = 0) ifTrue:[DialogView warn: '  
  Sie haben in der Excel-Tabelle für die Simulation eine Startzeit gewählt,  
  die auf einen arbeitsfreien Tag fällt. Das Feld ist in Excel im Tabellenblatt Aufträge (oben)  
  rot hinterlegt.  
  Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE  
  anschließend neu.'. self terminate.].
```

"Evtl. vorzeitigen Stop Simulation lesen, prüfen ob arbeitsfreier Tag"

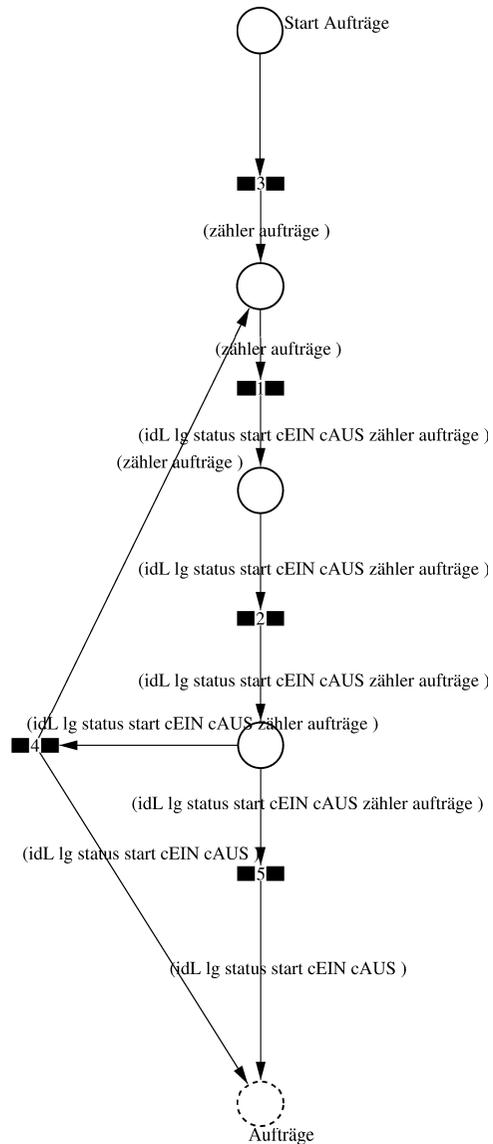
```
stop:= (DDE getData: Kanal2 item:'Z1S9') asNumber.  
stoppFrei:= (DDE getData: Kanal2 item:'Z1S10') asNumber.  
(stoppFrei = 0) ifTrue:[DialogView warn: '  
  Sie haben in der Excel-Tabelle für die Simulation eine Abbruchzeit gewählt, die auf einen  
  arbeitsfreien Tag fällt.  
  Das Feld ist in Excel rot hinterlegt.
```

Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE
anschließend neu.'. self terminate.].

"Marke in Auftragspool einlegen"

```
startAufträge:= self placeNamed:'Start Aufträge'.  
self addTokenTo: startAufträge.
```

Module: Auftragspool



====> Development Comment <====

Modul: Auftragspool
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Auftragspool

Aufgabe: In diesem Modul "Auftragspool" werden die Aufträge aus Excel als Lose eingelesen.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

 Aufträge idL Eindeutige Nummer des Loses, in Excel festgelegt.
 Aufträge lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
 Aufträge status Zeigt den aktuellen Status des Loses in der Simulation an.
 Aufträge start Gibt die Simulationszeit an, zu der das Los startet.
 Aufträge cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
 Aufträge cAUS Collection, welche die Auswerte-Daten des Loses speichert.

1: Transition

====> Transition Variables <====

#z
 #s
 #tB
 #tR
 #tRNetto
 #tBNetto
 #hilfsvariable
 #bST
 #tBVarProzent
 #tRVarProzent

#status
#freiStart
#freiFertig

====> Action Code <====

"Aus Excel eingelesen werden:

idL Eindeutige Nummer des Loses
lg Anzahl der Werkstücke in dem Los
start Startzeit des Loses
bST Bearbeitungsstation (wird für jede Bearbeitungsstation gesondert eingelesen)
tRNetto Rüstzeit je Los (wird für jede Bearbeitungsstation gesondert eingelesen)
tRVarProzent Schwankung der Rüstzeit (wird für jede Bearbeitungsstation gesondert eingelesen)
tBNetto Bearbeitungszeit je Werkstück (wird für jede Bearbeitungsstation gesondert eingelesen)
tBVarProzent Schwankung der Bearbeitungszeit (wird für jede Bearbeitungsstation gesondert eingelesen)"

"Erzeugt werden:

status Zeigt den aktuellen Status des Loses in der Simulation an
cEIN Collection, welche tB und tR speichert
cAUS Collection, welche die Auswerte-Daten des Loses speichert.
tB Collection, welche tBNetto und tBVar speichert
tR Collection, welche tRNetto und tRVar speichert"

z:= zähler.
s:= 49.

"Aufträge aus Excel einlesen"

idL:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:1.
lg:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:2.
status:= #eingelesen.
start:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:3.
freiStart:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:54.
freiFertig:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:55.
cEIN:= OrderedCollection new.

(freiStart = 99) ifTrue: [DialogView warn: '

Sie haben in der Excel-Tabelle bei Los ', z printString, ' keine Startzeit eingetragen. Das Feld ist im Tabellenblatt Aufträge in Excel rot hinterlegt.

Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

(freiStart = 0) ifTrue: [DialogView warn: '

Sie haben in der Excel-Tabelle bei Los ', z printString, ' eine Startzeit gewählt, die auf einen arbeitsfreien Tag fällt.

Das Feld ist im Tabellenblatt Aufträge in Excel rot hinterlegt.

Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

(freiFertig = 0) ifTrue: [

DialogView warn: '

Sie haben in der Excel-Tabelle bei Los ', z printString, ' für die Soll-Fertigstellung ein Datum gewählt, das auf einen arbeitsfreien Tag fällt.

Das Feld ist im Tabellenblatt Aufträge in Excel rot hinterlegt.

Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

10 timesRepeat:

[
tB:= OrderedCollection new.
tR:= OrderedCollection new.
tBVarProzent:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at: s+4.
"Schwankung Bearbeitungszeit"
tBNetto:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at: s+3.
"Bearbeitungszeit"
tRVarProzent:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at: s+2.
"Schwankung Rüstzeit"
tRNetto:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at: s+1.
"Rüstzeit"
hilfsvariable:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at: s.
"bST"

(hilfsvariable = 0)

ifTrue: [((tRNetto + tRVarProzent + tBNetto + tBVarProzent) ~= 0) ifTrue: [DialogView warn: '

Sie haben in der Excel-Tabelle bei Los ', z printString, ' zwar Rüst- bzw. Bearbeitungszeiten eingegeben, jedoch keine Bearbeitungsstation ausgewählt.

Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

s:=s-5.

]

ifFalse:[

((hilfsvariable = 8) or: (hilfsvariable = 11))

ifTrue: [((tBNetto+tBVarProzent) ~=0) ifTrue: [DialogView warn: '

Sie haben in der Excel-Tabelle bei Los ', z printString, ' die Bearbeitungsstation

Gleitschleifmaschine oder Extern ausgewählt.

Bitte geben Sie dort die gesamte Bearbeitungszeit des Loses unter der Rüstzeit ein, die Bearbeitungszeit/Stück bleibt frei.

Bitte korrigieren Sie diese Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

```
]
ifFalse: [((tBNetto = 0) and: (tBVarProzent~=0)) ifTrue: [DialogView warn: '
Sie haben in der Excel-Tabelle bei Los ', z printString, ' zwar eine Schwankung für die
Bearbeitungszeit eingegeben,
jedoch keine Bearbeitungszeit.
```

Bitte korrigieren Sie diese Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

```
]
((tRNetto = 0) and: (tRVarProzent~=0)) ifTrue: [DialogView warn: '
Sie haben in der Excel-Tabelle bei Los ', z printString, ' zwar eine Schwankung für die
Rüstzeit eingegeben, jedoch keine Rüstzeit.
```

Bitte korrigieren Sie diese Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

```
((tBNetto + tBVarProzent + tRNetto + tRVarProzent) =0)
ifTrue: [DialogView warn: '
Sie haben in der Excel-Tabelle bei Los ', z printString, ' zwar eine Bearbeitungsstation
ausgewählt, jedoch keine Rüst- bzw. Bearbeitungszeiten eingegeben.
```

Bitte prüfen und korrigieren Sie diese Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

```
ifFalse: [tR addFirst: tRVarProzent.
tR addFirst: tRNetto.
tB addFirst: tBVarProzent.
tB addFirst: tBNetto.
cEIN addFirst: tB.
cEIN addFirst: tR.
hilfsvariable = 0 ifTrue: [bST:= 0].
hilfsvariable = 1 ifTrue: [bST:= #cncDreh].
hilfsvariable = 2 ifTrue: [bST:= #cncFräs].
hilfsvariable = 3 ifTrue: [bST:= #cncDrehFräs].
hilfsvariable = 4 ifTrue: [bST:= #manDreh].
hilfsvariable = 5 ifTrue: [bST:= #manFräs].
hilfsvariable = 6 ifTrue: [bST:= #manBohr].
hilfsvariable = 7 ifTrue: [bST:= #manSchleif].
hilfsvariable = 8 ifTrue: [bST:= #gleitSchleif].
hilfsvariable = 9 ifTrue: [bST:= #montage].
hilfsvariable = 10 ifTrue: [bST:= #prüf].
hilfsvariable = 11 ifTrue: [bST:= #extern].
cEIN addFirst: bST.
```

```
]
s:=s-5.
].
].
cAUS:= Dictionary new.
```

2: Transition

====> Action Code <====

```
"Inhalte Container cAUS definieren"
cAUS at:#bearbSt put: OrderedCollection new.
cAUS at:#tEnde put:0.
cAUS at: #tBearb put: 0.
cAUS at:#tRüst put: 0.
cAUS at: #tWarte put: 0.
cAUS at: #tStart put: 0.
```

3: Transition

====> Transition Variables <====

```
#aufträge
#zähler
```

====> Action Code <====

```
"Auftragsanzahl aus Excel einlesen"
Kanal2:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_EIN'.
aufträge:= (DDE getData: Kanal2 item:'Z1S3') asNumber.
```

```
(aufträge=0) ifTrue: [DialogView warn: '
Sie haben in der Excel-Tabelle keine Aufträge eingetragen.
```

Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

```
"Zähler aktualisieren"
zähler:= 1.
```

4: Transition

====> Condition Code <====

```
zähler < aufträge
```

====> Action Code <====

"Zähler aktualisieren"
zähler:=zähler+1.

5: Transition

====> Transition Variables <====

#backControlSim

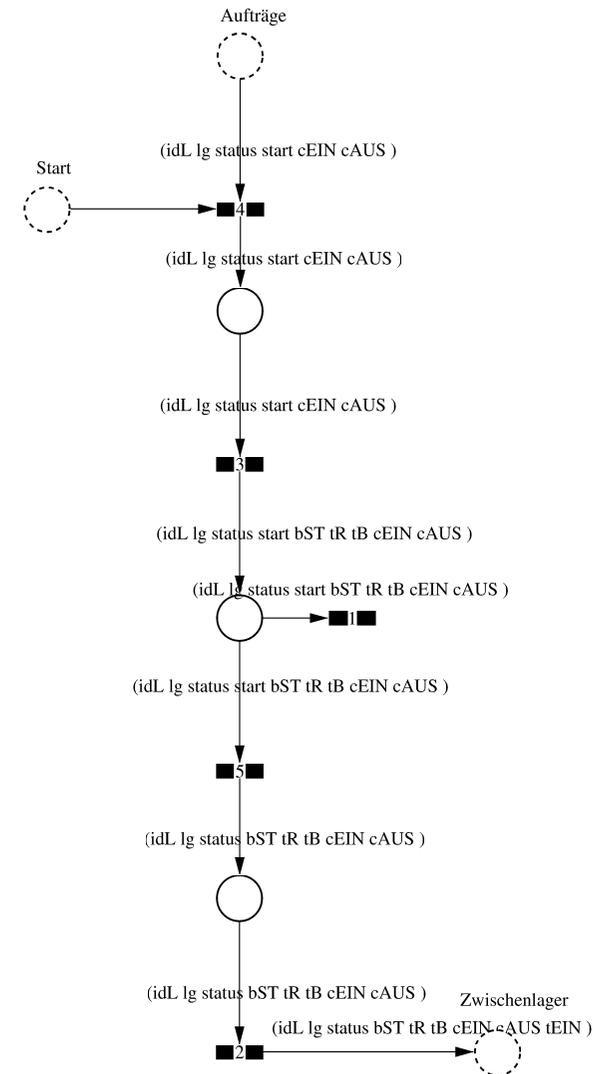
====> Condition Code <====

zähler =aufträge

====> Action Code <====

"Marke zurück ins Modul Control_Start legen"
backControlSim:= self placeNamed: 'Simulation starten'.
self addTokenTo: backControlSim.

Module: Arbeitsvorbereitung



====> Development Comment <====

Modul: Arbeitsvorbereitung
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Arbeitsvorbereitung

Aufgabe: In diesem Modul "Arbeitsvorbereitung" werden die Daten der Lose für die Simulation aufbereitet und die Lose in die Produktion gegeben.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Start -- --

Aufträge idL Eindeutige Nummer des Loses, in Excel festgelegt.
Aufträge lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Aufträge status Zeigt den aktuellen Status des Loses in der Simulation an.
Aufträge start Gibt die Simulationszeit an, zu der das Los startet.
Aufträge cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Aufträge cAUS Collection, welche die Auswerte-Daten des Loses speichert.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.

1: Transition

====> Condition Code <====

start < 0

2: Transition

====> Transition Variables <====

#tEIN

====> Action Code <====

"tEIN eintragen"
tEIN:= (CurrentTime) value.

3: Transition

====> Transition Variables <====

#tB
#bST

====> Action Code <====

"1. Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
if True:
[bST:= cEIN removeFirst.
tR:= cEIN removeFirst.
tB:= cEIN removeFirst.]
if False:
[bST:= #ende.
tR:= #ende.
tB:=#ende.].

4: Transition

====> Transition Variables <====

#container

====> Action Code <====

"Container ExcelMaschAus erzeugen und mit einer 0 füllen (löscht später alte Daten aus

Excel)"
ExcelMaschAus:= OrderedCollection new.
container:= OrderedCollection new.
container addFirst: 0.
ExcelMaschAus addFirst: container.

5: Transition

====> Condition Code <====

start >=0

====> Delay Code <====

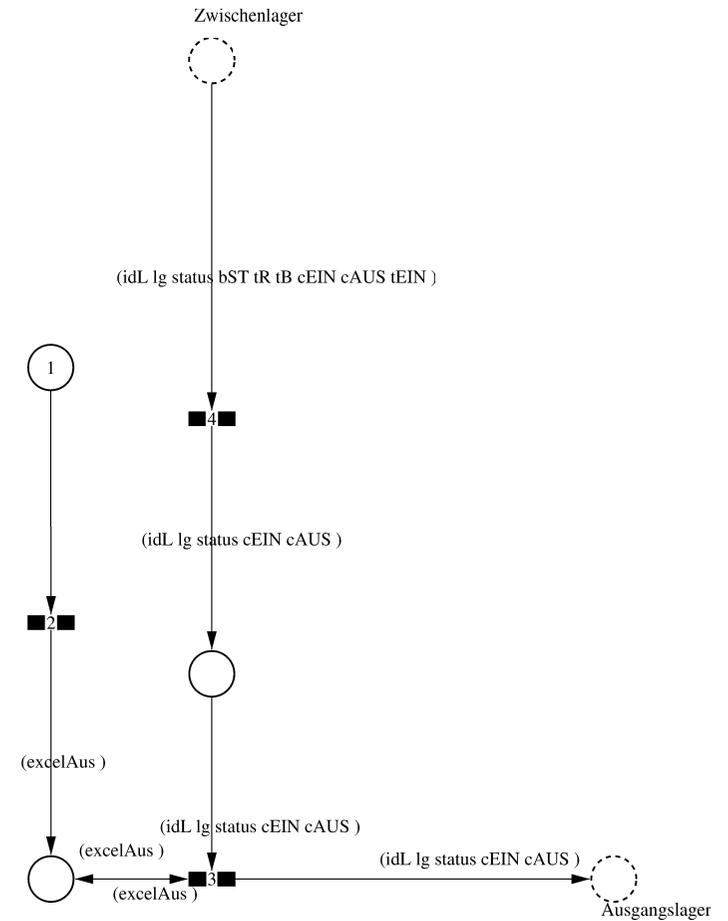
start

====> Action Code <====

"Status Los umsetzen"
status:= #wartet.

cAUS at: #tStart put: start.

Module: Arbeitsnachbereitung



====> Development Comment <====

Modul: Arbeitsnachbereitung
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Arbeitsnachbereitung

Aufgabe: In dem Modul "Arbeitsnachbereitung" werden die Daten der simulierten Lose sowie der Bearbeitungsstationen zurück in Excel-Datei geschrieben.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

Ausgangslager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Ausgangslager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Ausgangslager status Zeigt den aktuellen Status des Loses in der Simulation an.
Ausgangslager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Ausgangslager cAUS Collection, welche die Auswerte-Daten des Loses speichert.

1: Place

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

#excelAus
#container

====> Action Code <====

"Container erzeugen um Daten in Excel zu schreiben"
excelAus:= OrderedCollection new.
container:= OrderedCollection new.
container addFirst: 0.
excelAus addFirst: container.

3: Transition

====> Transition Variables <====

#tStart
#tWarte
#tRüst
#tBearb
#tEnde
#tDLZ.
#container
#a
#b
#bST
#tEIN
#bearbSt

====> Action Code <====

"Daten der Bearbeitungsstationen in Excel schreiben"
container:= OrderedCollection new.
bearbSt:= cAUS at: #bearbSt.
a:= bearbSt size.
b:= a/2.
b timesRepeat: [
bST:= bearbSt removeFirst.
tEIN:= bearbSt removeFirst.

```
container addFirst: tEIN.  
container addFirst: bST.  
].  
tDLZ:= cAUS at: #tDLZ.  
container addFirst: tDLZ.  
tEnde:= cAUS at: #tEnde.  
container addFirst: tEnde.  
tBearb:= cAUS at: #tBearb.  
container addFirst: tBearb.  
tRüst:= cAUS at: #tRüst.  
container addFirst: tRüst.  
tWarte:= cAUS at: #tWarte.  
container addFirst: tWarte.  
tStart:= cAUS at: #tStart.  
container addFirst: tStart.  
container addFirst: idL.  
excelAus add: container.  
Kanal1:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS'.  
DDE writeNumbersToExcel: Kanal1 startAt: 'Z3S1' dataList: excelAus.
```

4: Transition

====> Transition Variables <====

```
#bST  
#tDLZ  
#tStart  
#status
```

====> Condition Code <====

```
bST = #ende.
```

====> Action Code <====

```
"Durchlaufzeit in cAUS schreiben"  
cAUS at: #tEnde put:(CurrentTime) value.  
tStart:= cAUS at: #tStart.  
tDLZ:= ((CurrentTime) value) - tStart.  
cAUS at: #tDLZ put: tDLZ.
```

```
"Status Los umsetzen"  
status:= #fertig.
```

Anhang A7

Simulationsmodell Musterunternehmen
(Automatische Dokumentation aus PACE)

PACE model: Simulationsmodell Musterunternehmen

January 13, 2016 10:23:21 am

Modules

Simulationsmodell Musterunternehmen

Maschinenpark

CNC-Fräsmaschinen

- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation
- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation
- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation

Manuelle Schleifmaschinen

- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation

Gleitschleifmaschinen

- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation

- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation

Manuelle Bohrmaschinen

- Bearbeitungsstation
- Station rüsten

- Werkstücke bearbeiten
- Control_Bearbeitungsstation
- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation
- Montage
- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation
- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation
- CNC-Drehmaschinen
- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation
- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation
- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation
- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation
- CNC-Dreh-Fräszentren
- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation
- Prüfplätze
- Bearbeitungsstation
- Station rüsten
- Werkstücke bearbeiten
- Control_Bearbeitungsstation
- Externe Bearbeitung

Bearbeitungsstation

Station rüsten

Werkstücke bearbeiten

Control_Bearbeitungsstation

Control_Start

Auftragspool

Arbeitsvorbereitung

Arbeitsnachbereitung

Global Variables:

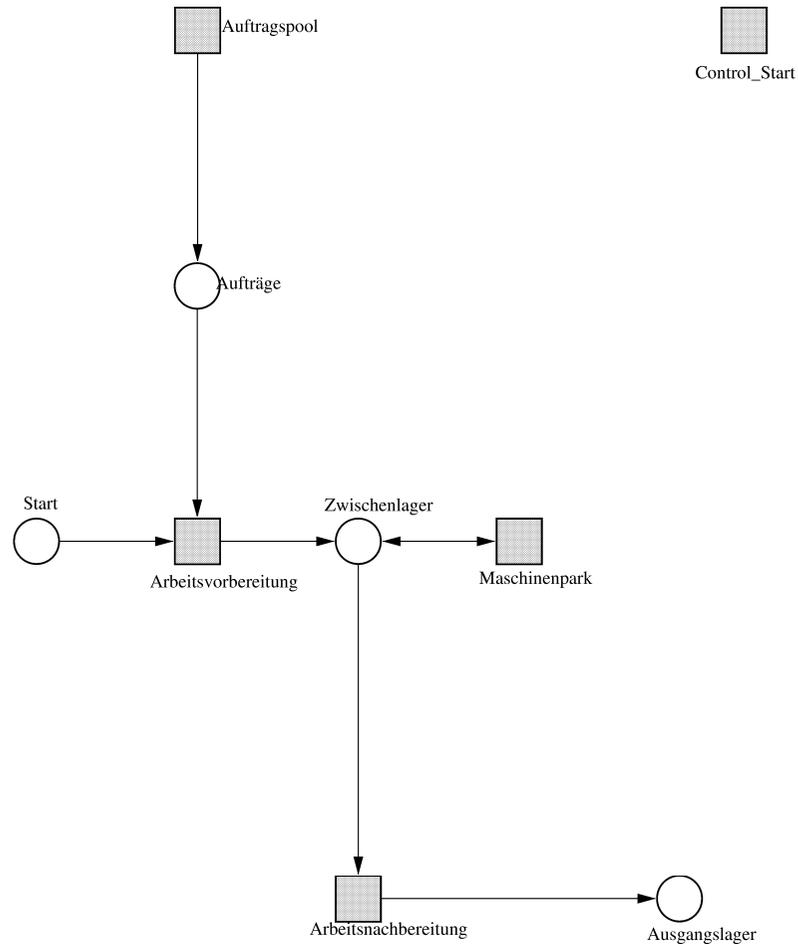
ExcelMaschAus

Kanal1

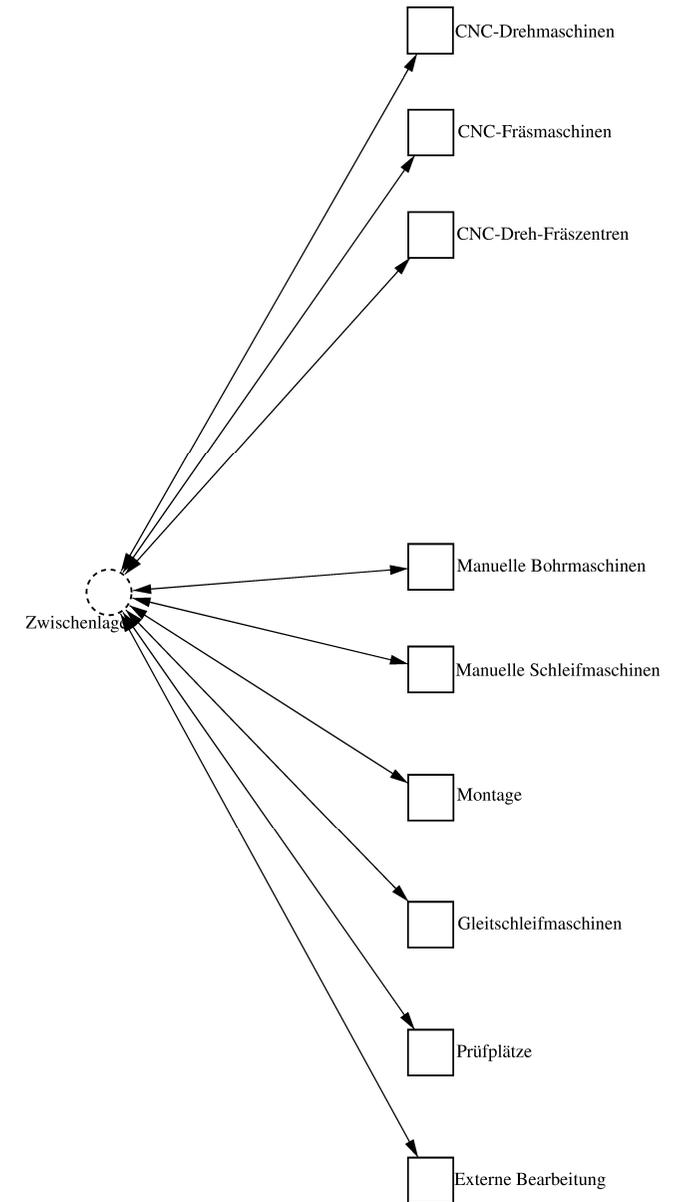
Kanal2

Kanal3

Module: Simulationsmodell Musterunternehmen



Module: Maschinenpark



====> Development Comment <====

Modul: Maschinenpark
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Maschinenpark

Aufgabe: Dieses Modul "Maschinenpark" beinhaltet die Bearbeitungsstationen.

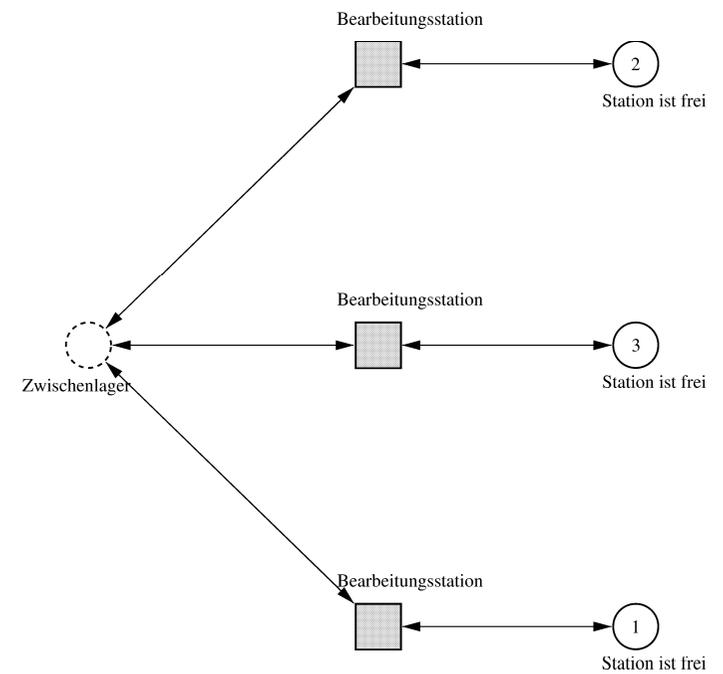
Anzahl der Eingangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Modul "Bearbeitungsstation".

Anzahl der Ausgangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Modul "Bearbeitungsstation".

Module: CNC-Fräsmaschinen



====> Development Comment <====

Modul: Kategorie Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Kategorie Bearbeitungsstation

Aufgabe: Dieses Modul "Kategorie Bearbeitungsstation" beinhaltet alle Bearbeitungsstationen EINES Typs (z.B. alle CNC-Drehmaschinen). Der Typ der Bearbeitungsstationen wird über die Modulvariable #maschine definiert. Die Anzahl der Bearbeitungsstationen in diesem Modul wird über die eingefügte Anzahl der Module "Bearbeitungsstation" eine Ebene tiefer definiert.

Anzahl der Eingangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Anzahl der Ausgangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Modulvariable Vorbesetzung (Initial) Beschreibung

#anzahl 0 Anzahl der Bearbeitungsstationen in diesem Modul.
#bearbeiten 0 Summiert die Bearbeitungszeiten der Bearbeitungsstationen während der Simulation auf.
#rüsten 0 Summiert die Rüstzeiten der Bearbeitungsstationen während der Simulation auf.
#station #eintragen Definition des Typs der Bearbeitungsstationen.
#warten 0 Summiert die Wartezeiten der Bearbeitungsstationen während der Simulation auf.

====> Net resp. Module Variables <====

#anzahl
#bearbeiten
#rüsten
#station

#warten

1: Place 'Station ist frei'

====> Initial Tokens <====
(CurrentTime)

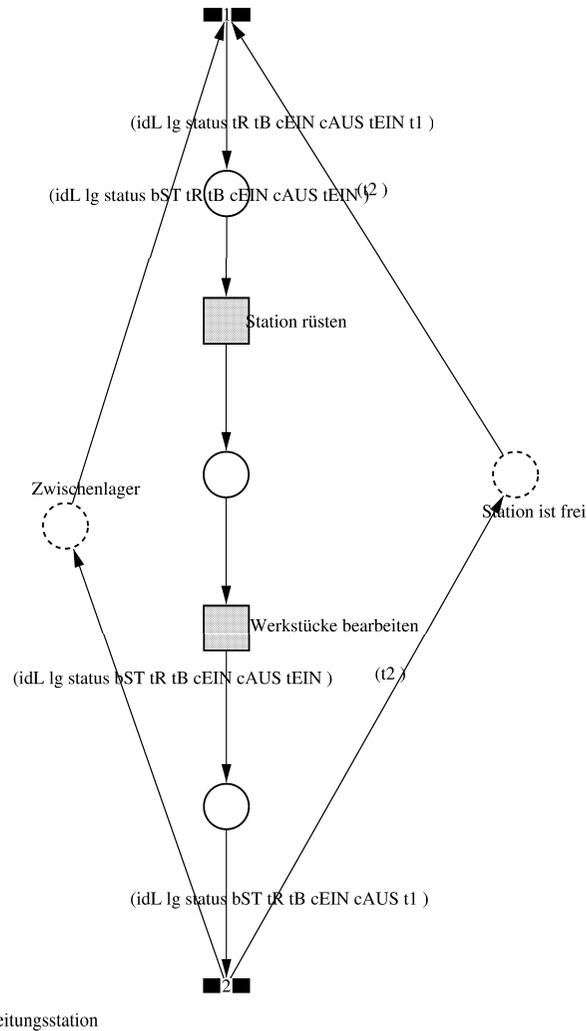
2: Place 'Station ist frei'

====> Initial Tokens <====
(CurrentTime)

3: Place 'Station ist frei'

====> Initial Tokens <====
(CurrentTime)

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	
Zwischenlager tEIN	Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.	
Station ist frei t2	Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.	

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel

====> Action Code <====

t2:= CurrentTime.
tEIN:= (CurrentTime) value.

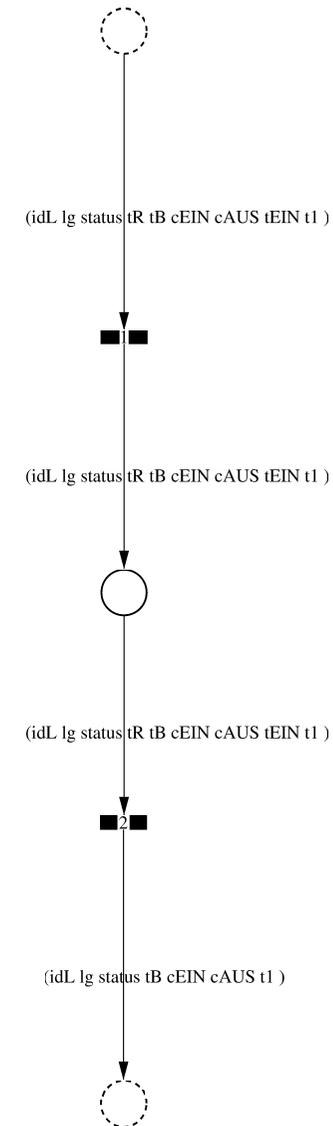
"Status Los umsetzen"
status:= #wartet.

"Status der Bearbeitungsstation in Excel schreiben"
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:= (((rüsten*10) rounded)//10)
printString, ', ', (((rüsten*10) rounded)\10)*10) printString.
bearbeitenExcel:= (((bearbeiten*10) rounded)//10)
printString, ', ', (((bearbeiten*10) rounded)\10)*10) printString.
wartenExcel:= (((warten*10) rounded)//10)
printString, ', ', (((warten*10) rounded)\10)*10) printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.  
(station =#manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].  
(station =#manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].  
(station =#montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"  
betriebszeit:= CurrentTime value.  
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)  
printString, ', ',(((betriebszeit*10)rounded)\10)*10)printString.  
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"

tRNetto:= tR removeFirst.

tRVarProzent:= tR removeFirst.

(tRVarProzent > 0) ifTrue:

[tRVar:= ((tRVarProzent/100) *tRNetto).

tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]

ifFalse:

[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"

tRüstAlt:= cAUS at: #tRüst.

tRüstHier:= ((CurrentTime) value) - tEIN.

tRüstNeu:= tRüstAlt + tRüstHier.

cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"

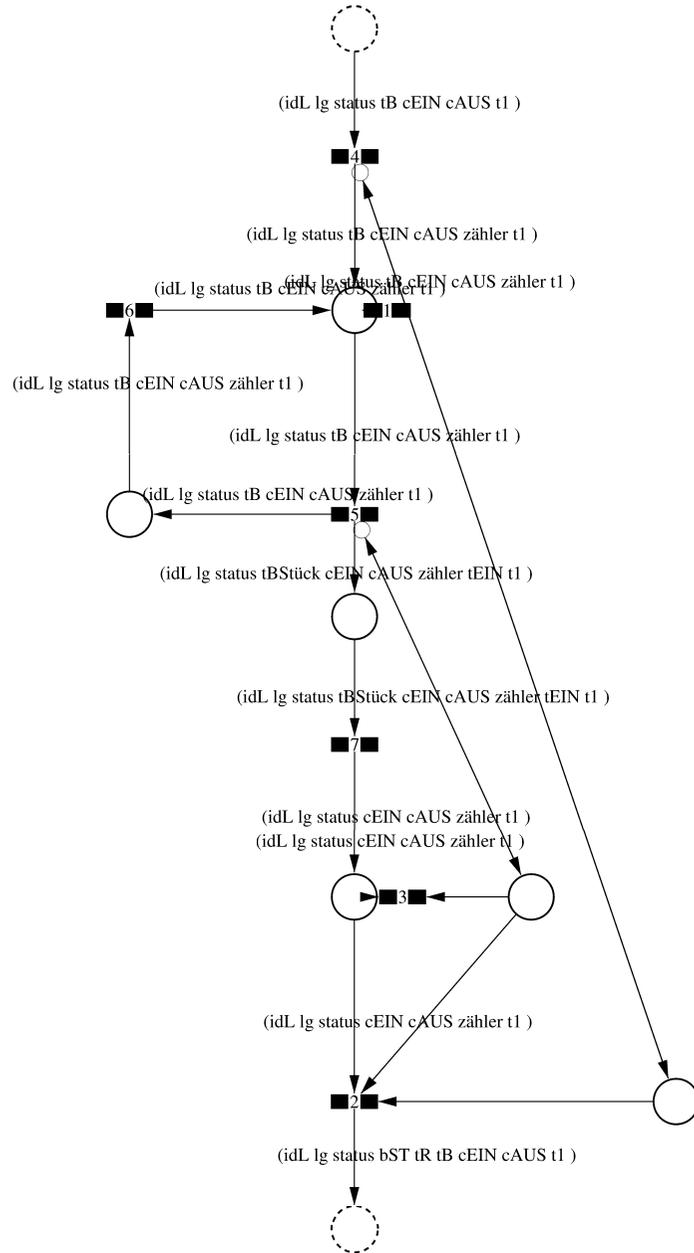
t2:= (CurrentTime) value.

rüstenAlt:= (self at: #rüsten) value.

rüstenNeu:= (t2-tEIN) +rüstenAlt.

(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
if True:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
if False:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"
zähler:= 1

5: Transition

====> Transition Variables <====

#zähler
#tBNetto
#tBVar
#tBStück
#tBVarProzent
#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"
tBNetto:= tB at: 1.
tBVarProzent:= tB at: 2.
(tBVarProzent > 0) if True:
 [tBVar:= ((tBVarProzent/100) *tBNetto).
 tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]
if False:
 [tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"
status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"
zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt
#tBearbHier
#tBearbNeu
#t2
#bearbeitenAlt
#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"
tBearbAlt:= cAUS at: #tBearb.
tBearbHier:= ((CurrentTime) value) - tEIN.
tBearbNeu:= tBearbAlt + tBearbHier.
cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
bearbeitenAlt:= (self at: #bearbeiten) value.
bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.
(self at: #bearbeiten) value: arbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.

Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"

anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

station:= (self at: #station) value.

Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.

(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu) printString.].

(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu) printString.].

(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu) printString.].

(station =#manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu) printString.].

(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu) printString.].

(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu) printString.].

(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu) printString.].

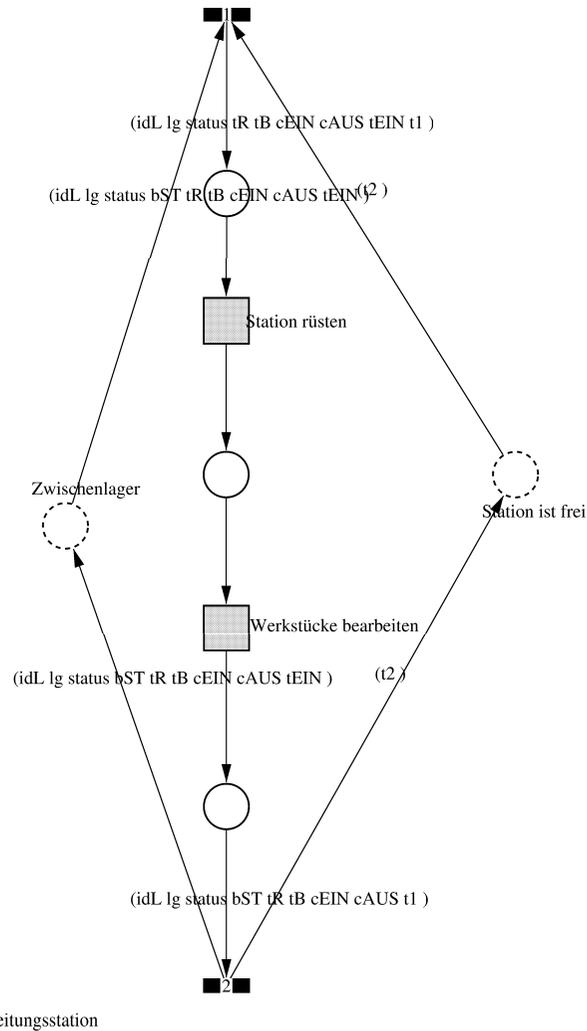
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu) printString.].

(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu) printString.].

(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu) printString.].

(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu) printString.].

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	
Zwischenlager tEIN	Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.	
Station ist frei t2	Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.	

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel

====> Action Code <====

t2:= CurrentTime.
tEIN:= (CurrentTime) value.

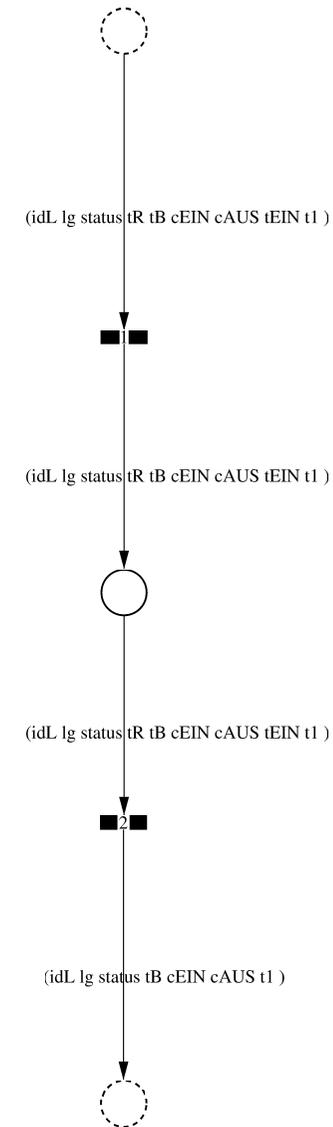
"Status Los umsetzen"
status:= #wartet.

"Status der Bearbeitungsstation in Excel schreiben"
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:= (((rüsten*10) rounded)//10)
printString, ', ', (((rüsten*10) rounded)\10)*10) printString.
bearbeitenExcel:= (((bearbeiten*10) rounded)//10)
printString, ', ', (((bearbeiten*10) rounded)\10)*10) printString.
wartenExcel:= (((warten*10) rounded)//10)
printString, ', ', (((warten*10) rounded)\10)*10) printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.  
(station =#manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].  
(station =#manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].  
(station =#montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"  
betriebszeit:= CurrentTime value.  
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)  
printString, ', ', (((betriebszeit*10) rounded)\10)*10)printString.  
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"

tRNetto:= tR removeFirst.

tRVarProzent:= tR removeFirst.

(tRVarProzent > 0) ifTrue:

[tRVar:= ((tRVarProzent/100) *tRNetto).

tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]

ifFalse:

[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"

tRüstAlt:= cAUS at: #tRüst.

tRüstHier:= ((CurrentTime) value) - tEIN.

tRüstNeu:= tRüstAlt + tRüstHier.

cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"

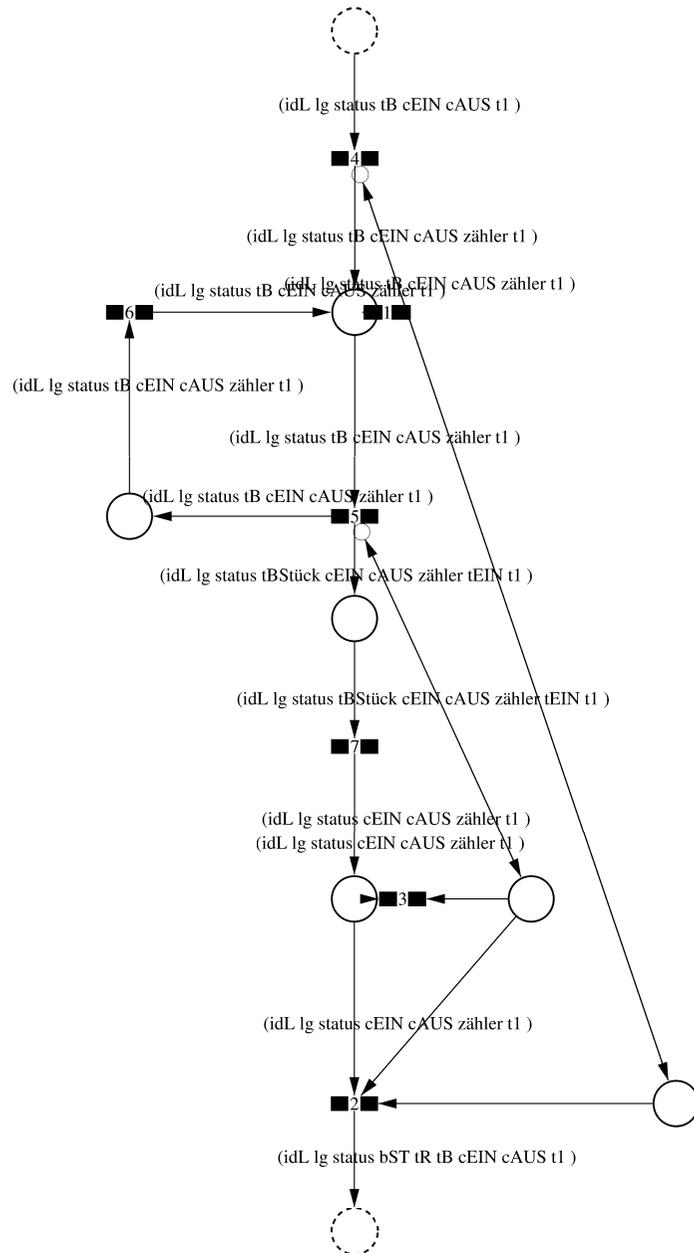
t2:= (CurrentTime) value.

rüstenAlt:= (self at: #rüsten) value.

rüstenNeu:= (t2-tEIN) +rüstenAlt.

(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- | Bezeichner | Attribute | Beschreibung |
|------------|--|--------------|
| -- idL | Eindeutige Nummer des Loses, in Excel festgelegt. | |
| -- lg | Anzahl der Werkstücke in dem Los, in Excel festgelegt. | |
| -- status | Zeigt den aktuellen Status des Loses in der Simulation an. | |
| -- tB | Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an. | |
| -- cEIN | Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert. | |
| -- cAUS | Collection, welche die Auswerte-Daten des Loses speichert. | |
| -- t1 | Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt. | |

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- | Bezeichner | Attribute | Beschreibung |
|------------|--|--------------|
| -- idL | Eindeutige Nummer des Loses, in Excel festgelegt. | |
| -- lg | Anzahl der Werkstücke in dem Los, in Excel festgelegt. | |
| -- status | Zeigt den aktuellen Status des Loses in der Simulation an. | |
| -- bST | Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird. | |
| -- tR | Gibt die Rüstzeit des Loses für die Bearbeitungsstation an. | |
| -- tB | Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an. | |
| -- cEIN | Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert. | |
| -- cAUS | Collection, welche die Auswerte-Daten des Loses speichert. | |
| -- t1 | Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt. | |

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
if True:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
if False:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"
zähler:= 1

5: Transition

====> Transition Variables <====

#zähler
#tBNetto
#tBVar
#tBStück
#tBVarProzent
#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"
tBNetto:= tB at: 1.
tBVarProzent:= tB at: 2.
(tBVarProzent > 0) if True:
 [tBVar:= ((tBVarProzent/100) *tBNetto).
 tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]
if False:
 [tBStück:= tBNetto].

tEIN:= (currentTime) value.

"Status Los umsetzen"
status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"
zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt
#tBearbHier
#tBearbNeu
#t2
#bearbeitenAlt
#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"
tBearbAlt:= cAUS at: #tBearb.
tBearbHier:= ((CurrentTime) value) - tEIN.
tBearbNeu:= tBearbAlt + tBearbHier.
cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
bearbeitenAlt:= (self at: #bearbeiten) value.
bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.
(self at: #bearbeiten) value: arbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.

Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

#anzahlAlt
#anzahlNeu
#station

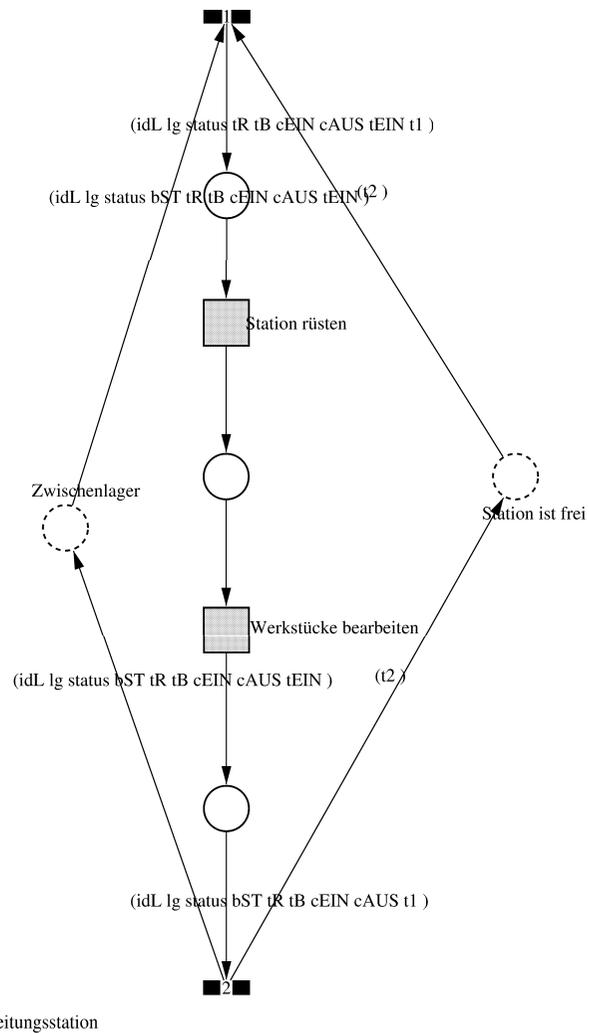
====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"

anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station = #manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel

====> Action Code <====

t2:= CurrentTime.
tEIN:= (CurrentTime) value.

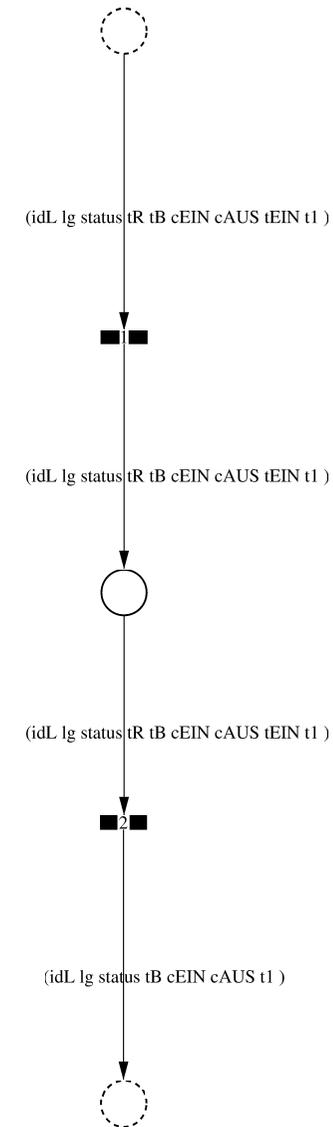
"Status Los umsetzen"
status:= #wartet.

"Status der Bearbeitungsstation in Excel schreiben"
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:= (((rüsten*10) rounded)//10)
printString, ', ', (((rüsten*10) rounded)\10)*10) printString.
bearbeitenExcel:= (((bearbeiten*10) rounded)//10)
printString, ', ', (((bearbeiten*10) rounded)\10)*10) printString.
wartenExcel:= (((warten*10) rounded)//10)
printString, ', ', (((warten*10) rounded)\10)*10) printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].  
(station =#manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].  
(station =#manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].  
(station =#montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"  
betriebszeit:= CurrentTime value.  
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)  
printString, ', ',(((betriebszeit*10)rounded)\10)*10)printString.  
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"

tRNetto:= tR removeFirst.

tRVarProzent:= tR removeFirst.

(tRVarProzent > 0) ifTrue:

[tRVar:= ((tRVarProzent/100) *tRNetto).

tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]

ifFalse:

[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"

tRüstAlt:= cAUS at: #tRüst.

tRüstHier:= ((CurrentTime) value) - tEIN.

tRüstNeu:= tRüstAlt + tRüstHier.

cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"

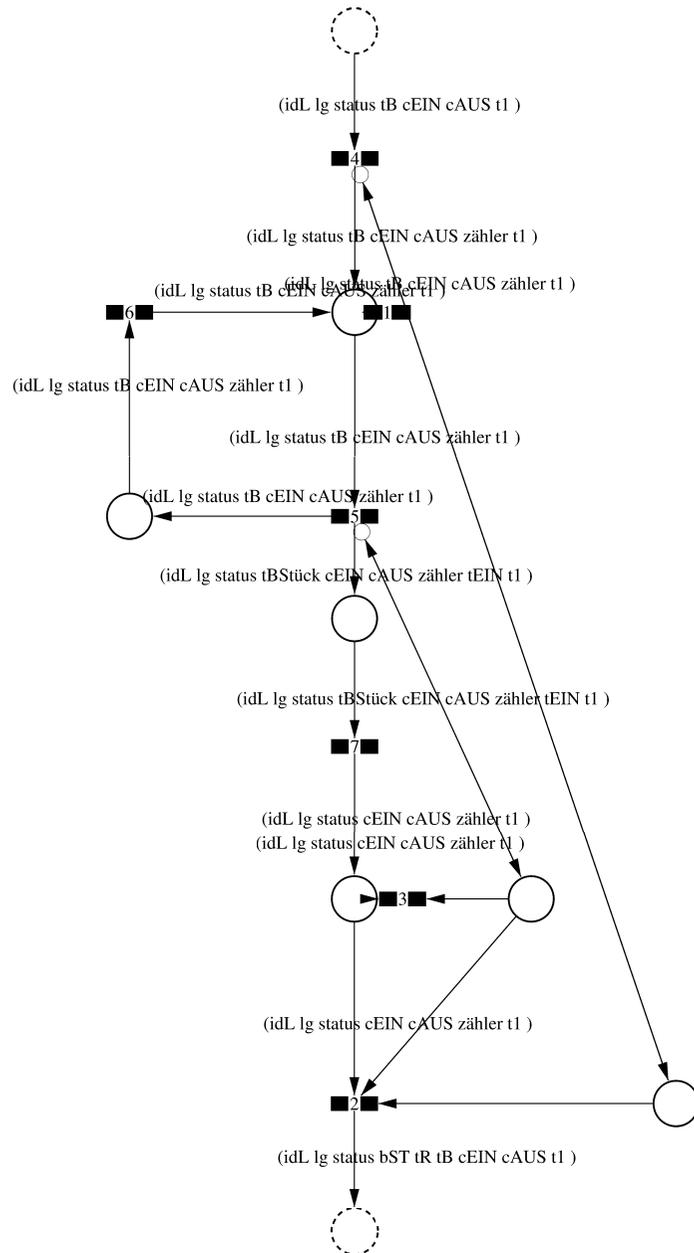
t2:= (CurrentTime) value.

rüstenAlt:= (self at: #rüsten) value.

rüstenNeu:= (t2-tEIN) +rüstenAlt.

(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
ifTrue:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
ifFalse:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"
zähler:= 1

5: Transition

====> Transition Variables <====

#zähler
#tBNetto
#tBVar
#tBStück
#tBVarProzent
#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"
tBNetto:= tB at: 1.
tBVarProzent:= tB at: 2.
(tBVarProzent > 0) ifTrue:
 [tBVar:= ((tBVarProzent/100) *tBNetto).
 tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]
ifFalse:
 [tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"
status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"
zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt
#tBearbHier
#tBearbNeu
#t2
#bearbeitenAlt
#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"
tBearbAlt:= cAUS at: #tBearb.
tBearbHier:= ((CurrentTime) value) - tEIN.
tBearbNeu:= tBearbAlt + tBearbHier.
cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
bearbeitenAlt:= (self at: #bearbeiten) value.
bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.
(self at: #bearbeiten) value: arbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.

Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

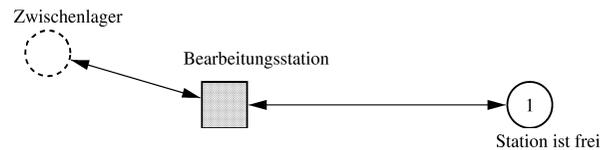
#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"
anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station = #manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Manuelle Schleifmaschinen



====> Development Comment <====

Modul: Kategorie Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Kategorie Bearbeitungsstation

Aufgabe: Dieses Modul "Kategorie Bearbeitungsstation" beinhaltet alle Bearbeitungsstationen EINES Typs (z.B. alle CNC-Drehmaschinen). Der Typ der Bearbeitungsstationen wird über die Modulvariable #maschine definiert. Die Anzahl der Bearbeitungsstationen in diesem Modul wird über die eingefügte Anzahl der Module "Bearbeitungsstation" eine Ebene tiefer definiert.

Anzahl der Eingangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Anzahl der Ausgangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Modulvariable Vorbesetzung (Initial) Beschreibung

#anzahl 0 Anzahl der Bearbeitungsstationen in diesem Modul.
#bearbeiten 0 Summiert die Bearbeitungszeiten der Bearbeitungsstationen während der Simulation auf.
#rüsten 0 Summiert die Rüstzeiten der Bearbeitungsstationen während der Simulation auf.
#station #eintragen Definition des Typs der Bearbeitungsstationen.
#warten 0 Summiert die Wartezeiten der Bearbeitungsstationen während der Simulation auf.

====> Net resp. Module Variables <====

#anzahl
#bearbeiten
#rüsten
#station

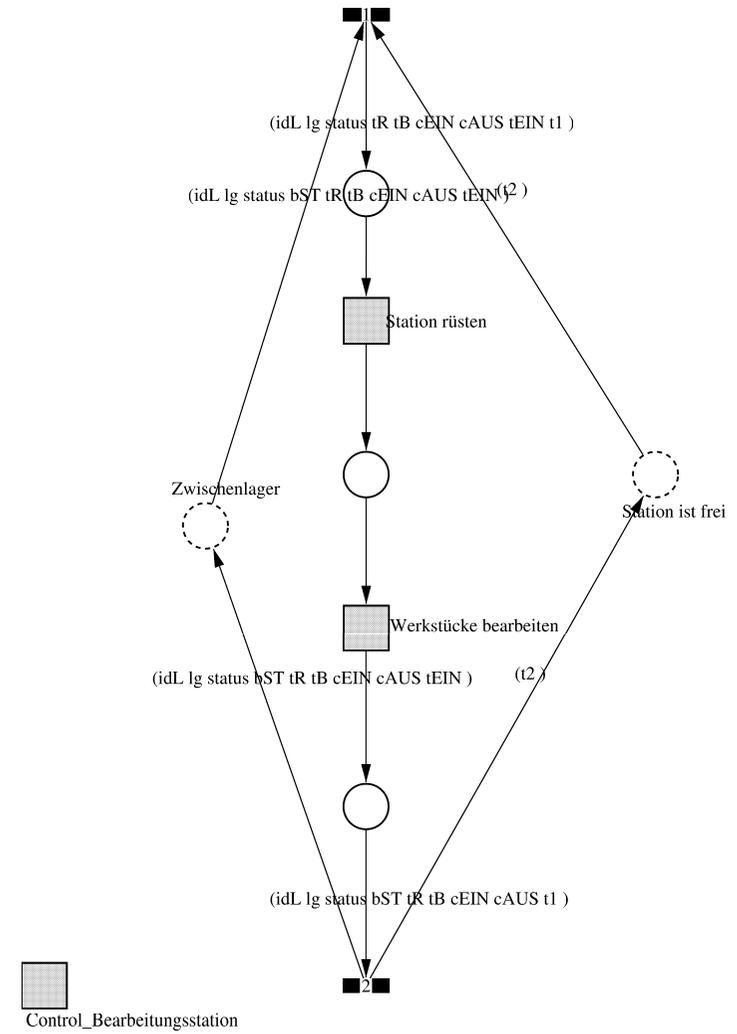
#warten

1: Place 'Station ist frei'

====> Initial Tokens <====

(CurrentTime)

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

```
#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel
```

```
====> Action Code <====
```

```
t2:= CurrentTime.
tEIN:= (CurrentTime) value.
```

```
"Status Los umsetzen"
status:= #wartet.
```

```
"Status der Bearbeitungsstation in Excel schreiben"
```

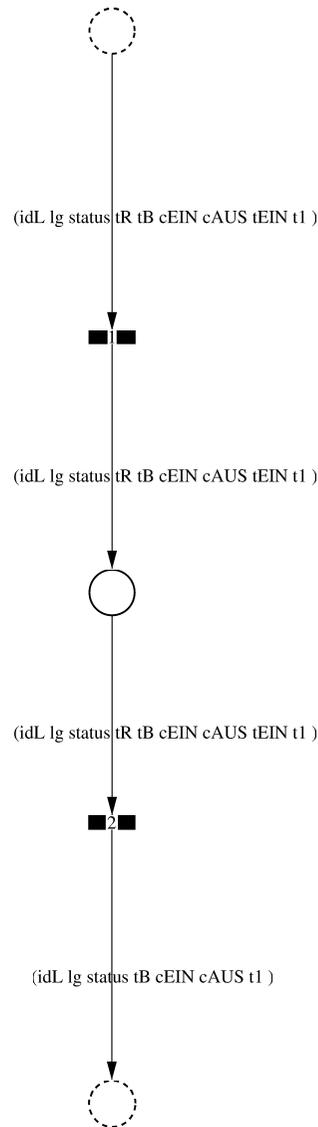
```
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:=(((rüsten*10) rounded)//10)
printString,',',(((rüsten*10)rounded)\10)*10)printString.
bearbeitenExcel:=(((bearbeiten*10) rounded)//10)
printString,',',(((bearbeiten*10)rounded)\10)*10)printString.
wartenExcel:=(((warten*10) rounded)//10)
printString,',',(((warten*10)rounded)\10)*10)printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.
```

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].
(station = #manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].
(station = #manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].
(station = #gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].
(station = #montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].
(station = #prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].
(station = #extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"
```

```
betriebszeit:= CurrentTime value.
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)
printString,',',(((betriebszeit*10)rounded)\10)*10)printString.
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

```
#tRNetto
#tR
#tRVar
#tRVarProzent
```

====> Action Code <====

```
"Rüstzeit und Schwankung auslesen und ausrechnen"
tRNetto:= tR removeFirst.
tRVarProzent:= tR removeFirst.
(tRVarProzent > 0) ifTrue:
[tRVar:= ((tRVarProzent/100) *tRNetto).
tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]
ifFalse:
[tR:= tRNetto].
```

2: Transition

====> Transition Variables <====

```
#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2
```

====> Delay Code <====

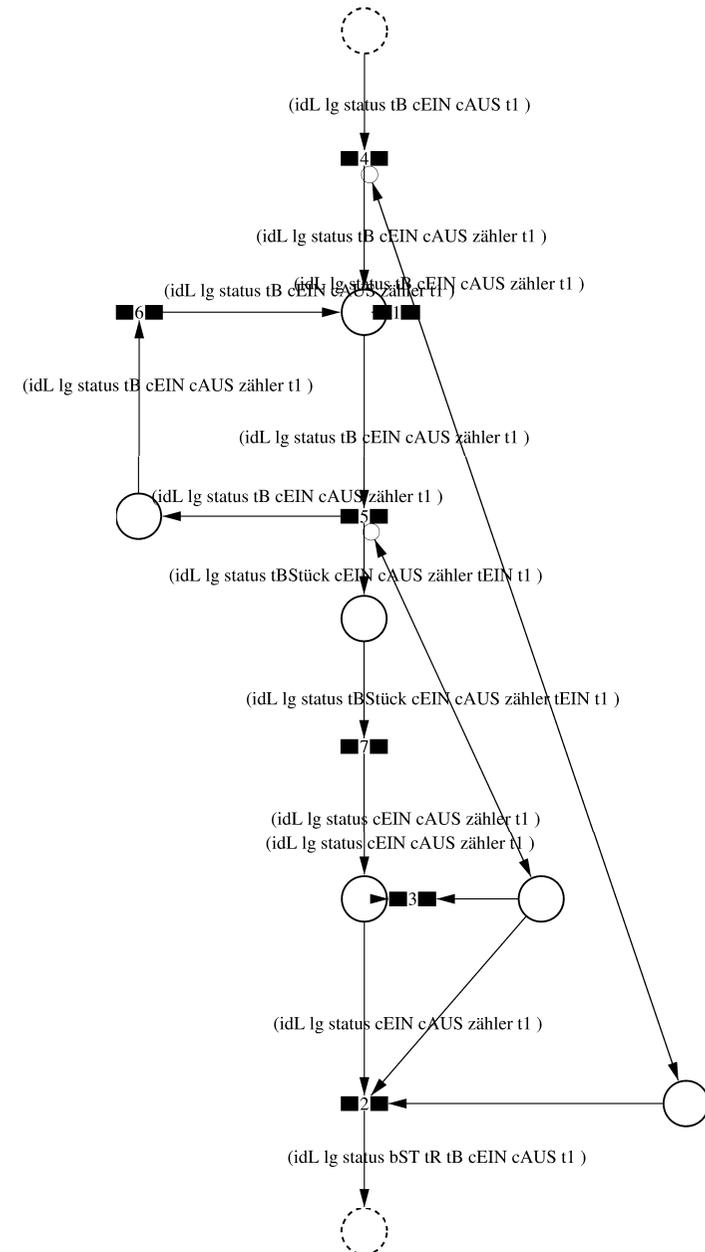
```
tR
```

====> Action Code <====

```
"Rüstzeit in cAUS schreiben"
tRüstAlt:= cAUS at: #tRüst.
tRüstHier:= ((CurrentTime) value) - tEIN.
tRüstNeu:= tRüstAlt + tRüstHier.
cAUS at: #tRüst put: tRüstNeu.
```

```
"#rüsten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
rüstenAlt:= (self at: #rüsten) value.
rüstenNeu:= (t2-tEIN) +rüstenAlt.
(self at: #rüsten) value: rüstenNeu.
```

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
ifTrue:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
ifFalse:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"

zähler:= 1

5: Transition

====> Transition Variables <====

#zähler

#tBNetto

#tBVar

#tBStück

#tBVarProzent

#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"

tBNetto:= tB at: 1.

tBVarProzent:= tB at: 2.

(tBVarProzent > 0) ifTrue:

[tBVar:= ((tBVarProzent/100) *tBNetto).

tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]

iffalse:

[tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"

status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"

zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt

#tBearbHier

#tBearbNeu

#t2

#bearbeitenAlt

#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"

tBearbAlt:= cAUS at: #tBearb.

tBearbHier:= ((CurrentTime) value) - tEIN.

tBearbNeu:= tBearbAlt + tBearbHier.

cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"

t2:= (CurrentTime) value.

bearbeitenAlt:= (self at: #bearbeiten) value.

bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.

(self at: #bearbeiten) value: bearbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



2

====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.
Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

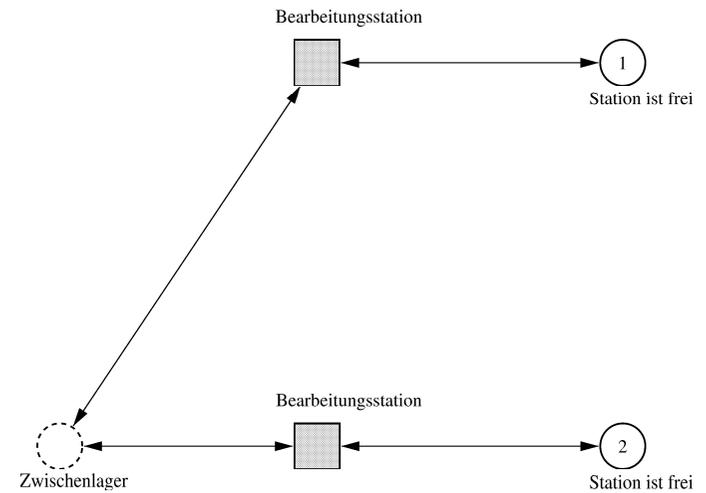
#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariabler auslesen, aktualisieren und in Excel schreiben"
anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station =#manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Gleitschleifmaschinen



====> Development Comment <====

Modul: Kategorie Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Kategorie Bearbeitungsstation

Aufgabe: Dieses Modul "Kategorie Bearbeitungsstation" beinhaltet alle Bearbeitungsstationen EINES Typs (z.B. alle CNC-Drehmaschinen). Der Typ der Bearbeitungsstationen wird über die Modulvariable #maschine definiert. Die Anzahl der Bearbeitungsstationen in diesem Modul wird über die eingefügte Anzahl der Module "Bearbeitungsstation" eine Ebene tiefer definiert.

Anzahl der Eingangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Anzahl der Ausgangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Modulvariable Vorbesetzung (Initial) Beschreibung

#anzahl 0 Anzahl der Bearbeitungsstationen in diesem Modul.
#bearbeiten 0 Summiert die Bearbeitungszeiten der Bearbeitungsstationen während der Simulation auf.
#rüsten 0 Summiert die Rüstzeiten der Bearbeitungsstationen während der Simulation auf.
#station #eintragen Definition des Typs der Bearbeitungsstationen.
#warten 0 Summiert die Wartezeiten der Bearbeitungsstationen während der Simulation auf.

====> Net resp. Module Variables <====

#anzahl
#bearbeiten
#rüsten
#station

#warten

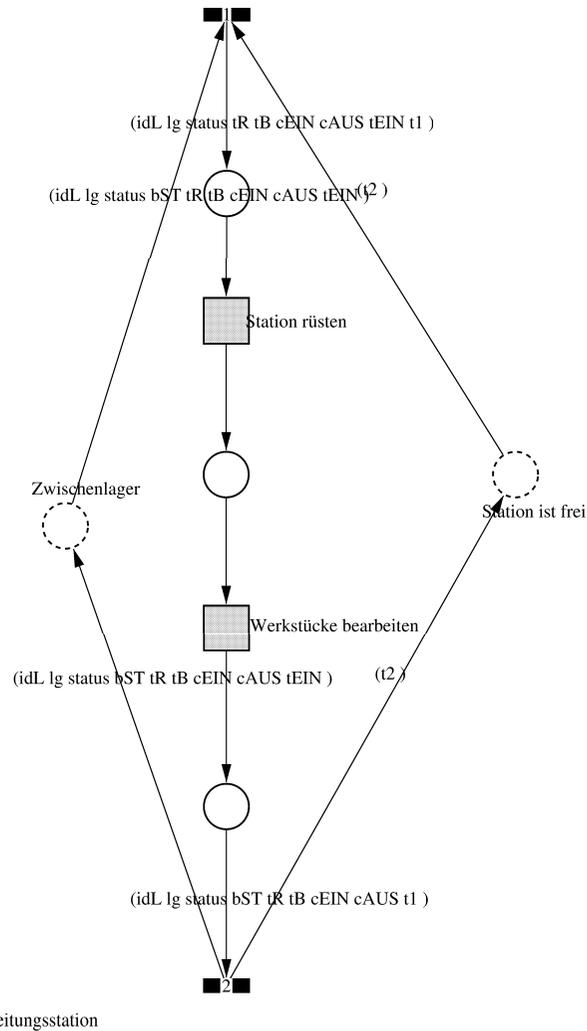
1: Place 'Station ist frei'

====> Initial Tokens <====
(CurrentTime)

2: Place 'Station ist frei'

====> Initial Tokens <====
(CurrentTime)

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	
Zwischenlager tEIN	Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.	
Station ist frei t2	Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.	

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel

====> Action Code <====

t2:= CurrentTime.
tEIN:= (CurrentTime) value.

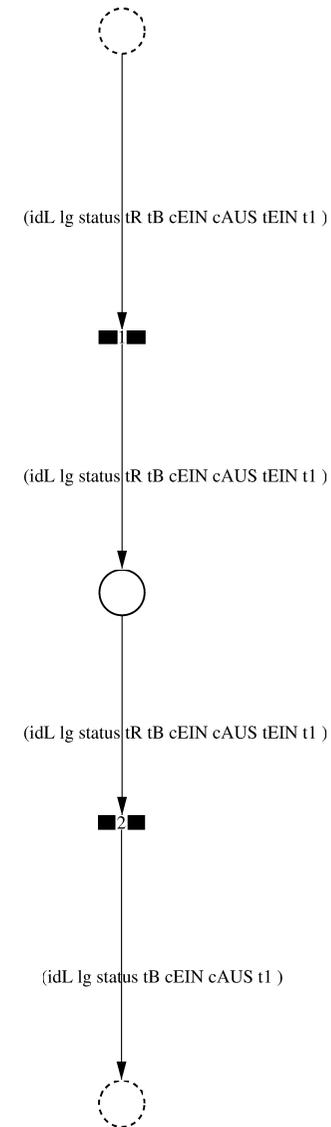
"Status Los umsetzen"
status:= #wartet.

"Status der Bearbeitungsstation in Excel schreiben"
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:= (((rüsten*10) rounded)//10)
printString, ', ', (((rüsten*10) rounded)\10)*10) printString.
bearbeitenExcel:= (((bearbeiten*10) rounded)//10)
printString, ', ', (((bearbeiten*10) rounded)\10)*10) printString.
wartenExcel:= (((warten*10) rounded)//10)
printString, ', ', (((warten*10) rounded)\10)*10) printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.  
(station =#manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].  
(station =#manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].  
(station =#montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"  
betriebszeit:= CurrentTime value.  
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)  
printString, ', ',(((betriebszeit*10)rounded)\10)*10)printString.  
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"

tRNetto:= tR removeFirst.

tRVarProzent:= tR removeFirst.

(tRVarProzent > 0) ifTrue:

[tRVar:= ((tRVarProzent/100) *tRNetto).

tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]

ifFalse:

[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"

tRüstAlt:= cAUS at: #tRüst.

tRüstHier:= ((CurrentTime) value) - tEIN.

tRüstNeu:= tRüstAlt + tRüstHier.

cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"

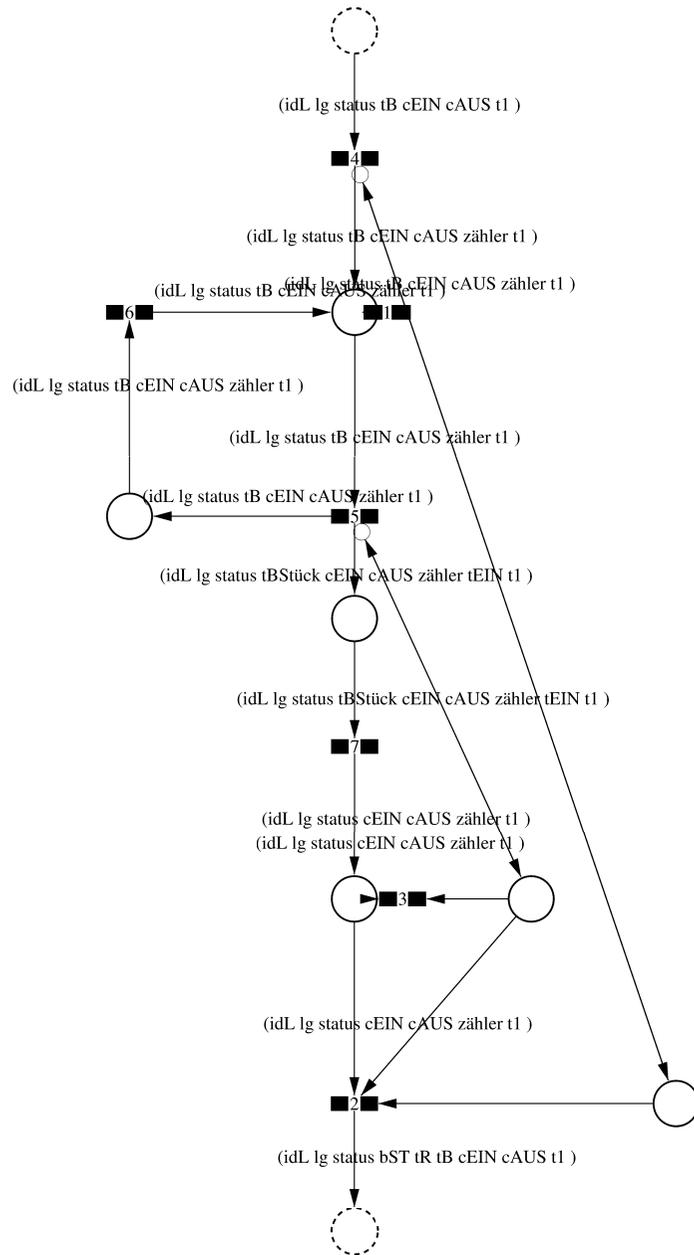
t2:= (CurrentTime) value.

rüstenAlt:= (self at: #rüsten) value.

rüstenNeu:= (t2-tEIN) +rüstenAlt.

(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- | Bezeichner | Attribute | Beschreibung |
|------------|--|--------------|
| -- idL | Eindeutige Nummer des Loses, in Excel festgelegt. | |
| -- lg | Anzahl der Werkstücke in dem Los, in Excel festgelegt. | |
| -- status | Zeigt den aktuellen Status des Loses in der Simulation an. | |
| -- tB | Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an. | |
| -- cEIN | Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert. | |
| -- cAUS | Collection, welche die Auswerte-Daten des Loses speichert. | |
| -- t1 | Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt. | |

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- | Bezeichner | Attribute | Beschreibung |
|------------|--|--------------|
| -- idL | Eindeutige Nummer des Loses, in Excel festgelegt. | |
| -- lg | Anzahl der Werkstücke in dem Los, in Excel festgelegt. | |
| -- status | Zeigt den aktuellen Status des Loses in der Simulation an. | |
| -- bST | Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird. | |
| -- tR | Gibt die Rüstzeit des Loses für die Bearbeitungsstation an. | |
| -- tB | Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an. | |
| -- cEIN | Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert. | |
| -- cAUS | Collection, welche die Auswerte-Daten des Loses speichert. | |
| -- t1 | Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt. | |

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
if True:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
if False:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"
zähler:= 1

5: Transition

====> Transition Variables <====

#zähler
#tBNetto
#tBVar
#tBStück
#tBVarProzent
#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"
tBNetto:= tB at: 1.
tBVarProzent:= tB at: 2.
(tBVarProzent > 0) if True:
 [tBVar:= ((tBVarProzent/100) *tBNetto).
 tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]
if False:
 [tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"
status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"
zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt
#tBearbHier
#tBearbNeu
#t2
#bearbeitenAlt
#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"
tBearbAlt:= cAUS at: #tBearb.
tBearbHier:= ((CurrentTime) value) - tEIN.
tBearbNeu:= tBearbAlt + tBearbHier.
cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
bearbeitenAlt:= (self at: #bearbeiten) value.
bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.
(self at: #bearbeiten) value: arbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.

Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

#anzahlAlt
#anzahlNeu
#station

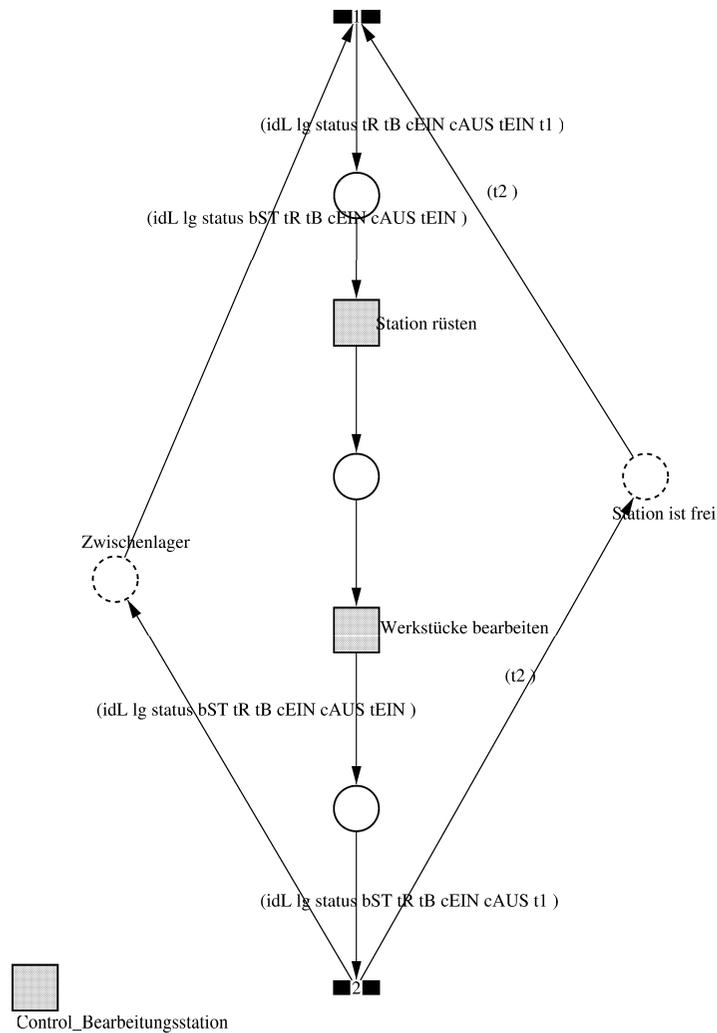
====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"

anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station = #manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	
Zwischenlager tEIN	Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.	
Station ist frei t2	Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.	

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel

====> Action Code <====

t2:= CurrentTime.
tEIN:= (CurrentTime) value.

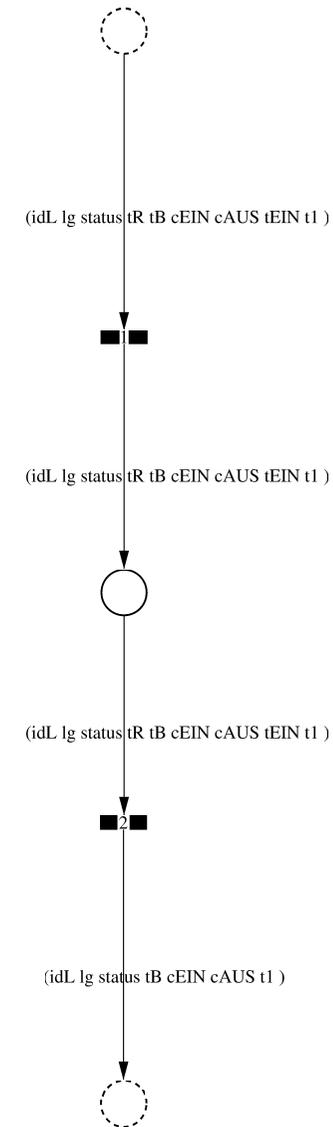
"Status Los umsetzen"
status:= #wartet.

"Status der Bearbeitungsstation in Excel schreiben"
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:= (((rüsten*10) rounded)//10)
printString, ', ', (((rüsten*10) rounded)\10)*10) printString.
bearbeitenExcel:= (((bearbeiten*10) rounded)//10)
printString, ', ', (((bearbeiten*10) rounded)\10)*10) printString.
wartenExcel:= (((warten*10) rounded)//10)
printString, ', ', (((warten*10) rounded)\10)*10) printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.  
(station =#manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].  
(station =#manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].  
(station =#montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"  
betriebszeit:= CurrentTime value.  
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)  
printString, ', ', (((betriebszeit*10) rounded)\10)*10)printString.  
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"

tRNetto:= tR removeFirst.

tRVarProzent:= tR removeFirst.

(tRVarProzent > 0) ifTrue:

[tRVar:= ((tRVarProzent/100) *tRNetto).

tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]

ifFalse:

[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"

tRüstAlt:= cAUS at: #tRüst.

tRüstHier:= ((CurrentTime) value) - tEIN.

tRüstNeu:= tRüstAlt + tRüstHier.

cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"

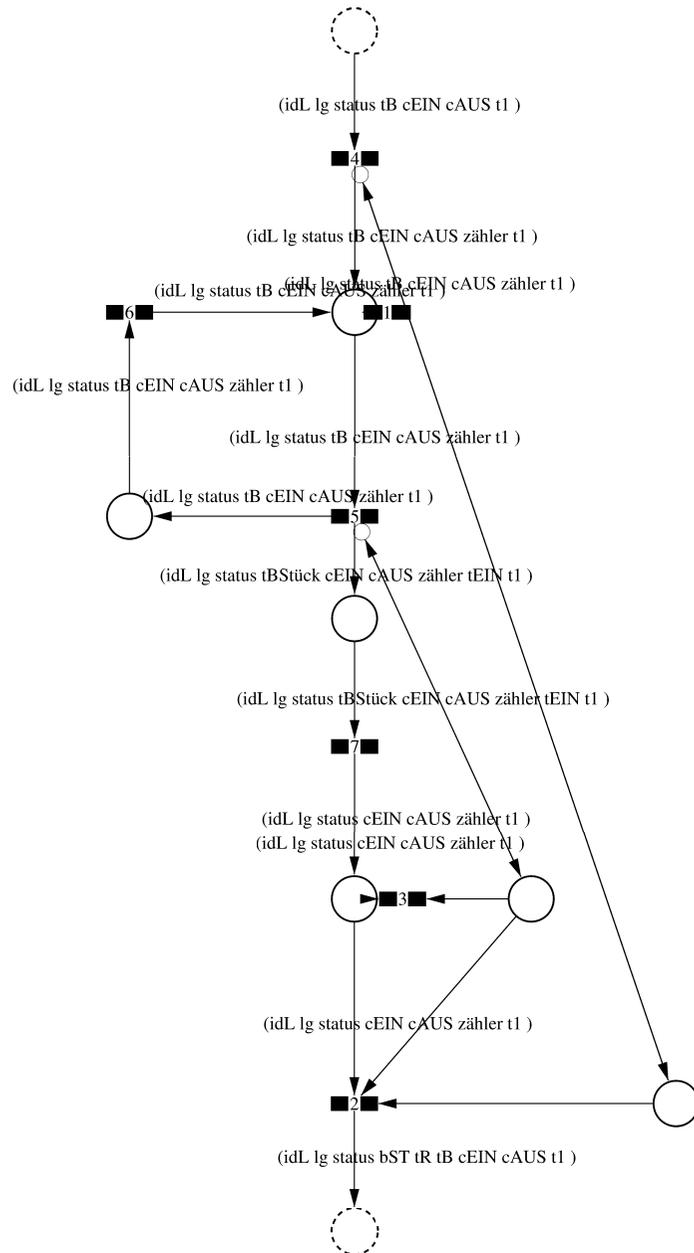
t2:= (CurrentTime) value.

rüstenAlt:= (self at: #rüsten) value.

rüstenNeu:= (t2-tEIN) +rüstenAlt.

(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
if True:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
if False:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"
zähler:= 1

5: Transition

====> Transition Variables <====

#zähler
#tBNetto
#tBVar
#tBStück
#tBVarProzent
#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"
tBNetto:= tB at: 1.
tBVarProzent:= tB at: 2.
(tBVarProzent > 0) if True:
 [tBVar:= ((tBVarProzent/100) *tBNetto).
 tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]
if False:
 [tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"
status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"
zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt
#tBearbHier
#tBearbNeu
#t2
#bearbeitenAlt
#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"
tBearbAlt:= cAUS at: #tBearb.
tBearbHier:= ((CurrentTime) value) - tEIN.
tBearbNeu:= tBearbAlt + tBearbHier.
cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
bearbeitenAlt:= (self at: #bearbeiten) value.
bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.
(self at: #bearbeiten) value: arbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.

Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

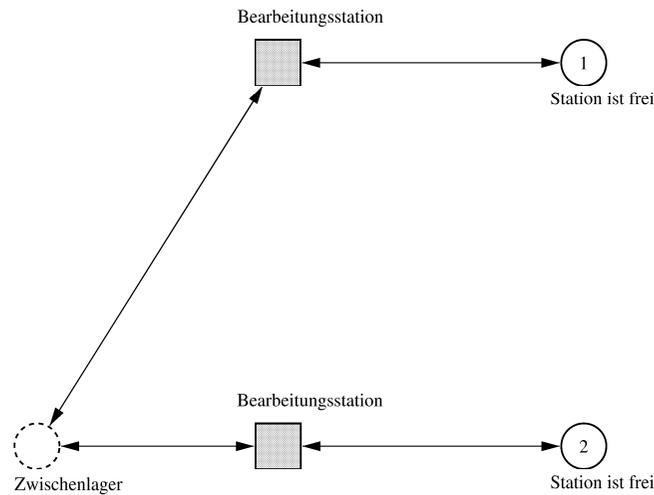
#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"
anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station = #manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Manuelle Bohrmaschinen



====> Development Comment <====

Modul: Kategorie Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Kategorie Bearbeitungsstation

Aufgabe: Dieses Modul "Kategorie Bearbeitungsstation" beinhaltet alle Bearbeitungsstationen EINES Typs (z.B. alle CNC-Drehmaschinen). Der Typ der Bearbeitungsstationen wird über die Modulvariable #maschine definiert. Die Anzahl der Bearbeitungsstationen in diesem Modul wird über die eingefügte Anzahl der Module "Bearbeitungsstation" eine Ebene tiefer definiert.

Anzahl der Eingangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Anzahl der Ausgangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Modulvariable Vorbesetzung (Initial) Beschreibung

#anzahl 0 Anzahl der Bearbeitungsstationen in diesem Modul.
#bearbeiten 0 Summiert die Bearbeitungszeiten der Bearbeitungsstationen während der Simulation auf.
#rüsten 0 Summiert die Rüstzeiten der Bearbeitungsstationen während der Simulation auf.
#station #eintragen Definition des Typs der Bearbeitungsstationen.
#warten 0 Summiert die Wartezeiten der Bearbeitungsstationen während der Simulation auf.

====> Net resp. Module Variables <====

#anzahl
#bearbeiten
#rüsten
#station

#warten

1: Place 'Station ist frei'

====> Initial Tokens <====

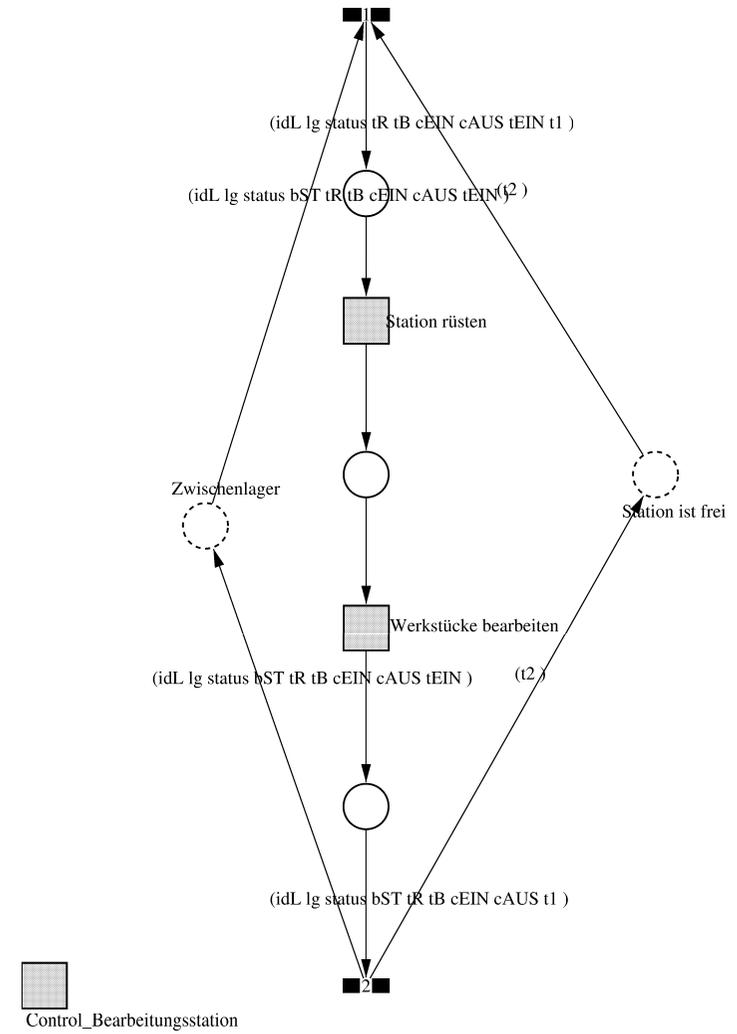
(CurrentTime)

2: Place 'Station ist frei'

====> Initial Tokens <====

(CurrentTime)

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

```
#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel
```

```
====> Action Code <====
```

```
t2:= CurrentTime.
tEIN:= (CurrentTime) value.
```

```
"Status Los umsetzen"
status:= #wartet.
```

```
"Status der Bearbeitungsstation in Excel schreiben"
```

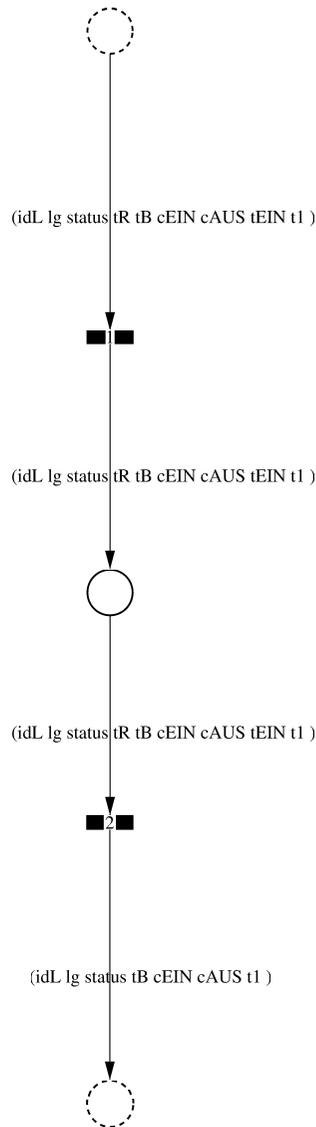
```
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:=(((rüsten*10) rounded)//10)
printString,',',(((rüsten*10)rounded)\10)*10)printString.
bearbeitenExcel:=(((bearbeiten*10) rounded)//10)
printString,',',(((bearbeiten*10)rounded)\10)*10)printString.
wartenExcel:=(((warten*10) rounded)//10)
printString,',',(((warten*10)rounded)\10)*10)printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.
```

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].
(station = #manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].
(station = #manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].
(station = #gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].
(station = #montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].
(station = #prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].
(station = #extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"
```

```
betriebszeit:= CurrentTime value.
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)
printString,',',(((betriebszeit*10)rounded)\10)*10)printString.
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

```
#tRNetto
#tR
#tRVar
#tRVarProzent
```

====> Action Code <====

```
"Rüstzeit und Schwankung auslesen und ausrechnen"
tRNetto:= tR removeFirst.
tRVarProzent:= tR removeFirst.
(tRVarProzent > 0) ifTrue:
[tRVar:= ((tRVarProzent/100) *tRNetto).
tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]
ifFalse:
[tR:= tRNetto].
```

2: Transition

====> Transition Variables <====

```
#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2
```

====> Delay Code <====

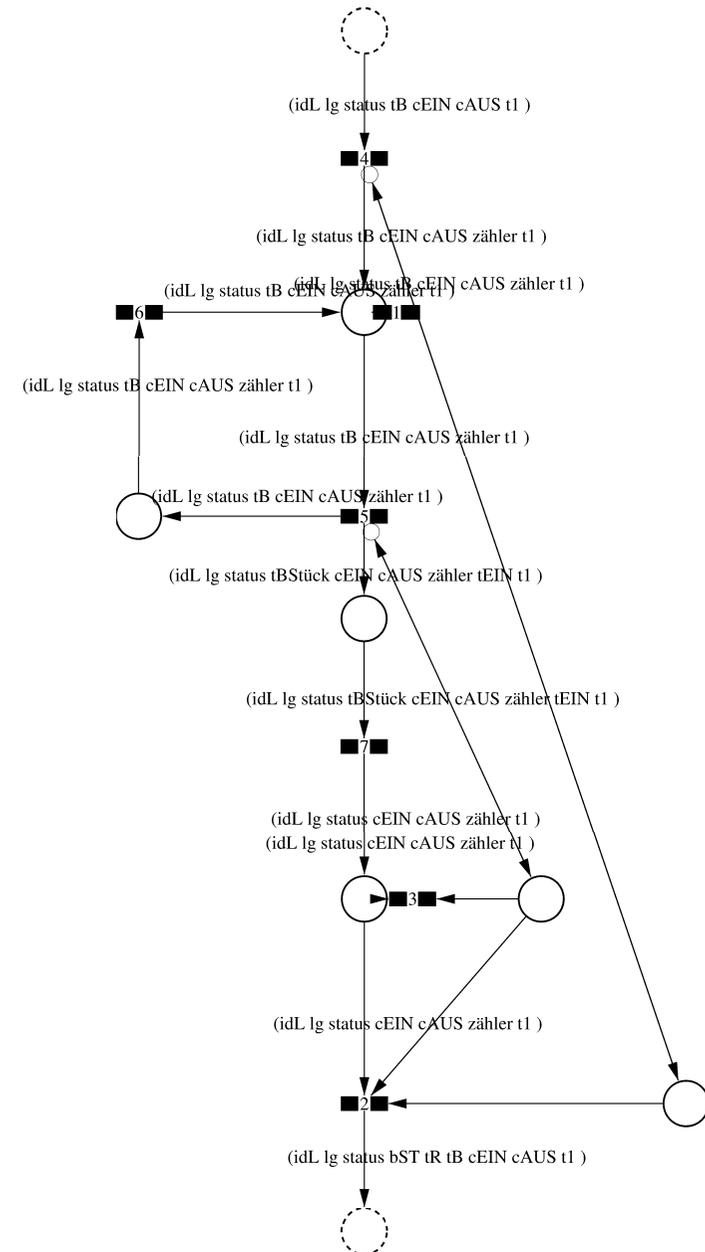
```
tR
```

====> Action Code <====

```
"Rüstzeit in cAUS schreiben"
tRüstAlt:= cAUS at: #tRüst.
tRüstHier:= ((CurrentTime) value) - tEIN.
tRüstNeu:= tRüstAlt + tRüstHier.
cAUS at: #tRüst put: tRüstNeu.
```

```
"#rüsten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
rüstenAlt:= (self at: #rüsten) value.
rüstenNeu:= (t2-tEIN) +rüstenAlt.
(self at: #rüsten) value: rüstenNeu.
```

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
ifTrue:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
ifFalse:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"

zähler:= 1

5: Transition

====> Transition Variables <====

#zähler

#tBNetto

#tBVar

#tBStück

#tBVarProzent

#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"

tBNetto:= tB at: 1.

tBVarProzent:= tB at: 2.

(tBVarProzent > 0) ifTrue:

[tBVar:= ((tBVarProzent/100) *tBNetto).

tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]

iffalse:

[tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"

status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"

zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt

#tBearbHier

#tBearbNeu

#t2

#bearbeitenAlt

#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"

tBearbAlt:= cAUS at: #tBearb.

tBearbHier:= ((CurrentTime) value) - tEIN.

tBearbNeu:= tBearbAlt + tBearbHier.

cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"

t2:= (CurrentTime) value.

bearbeitenAlt:= (self at: #bearbeiten) value.

bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.

(self at: #bearbeiten) value: bearbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



2

====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.
Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

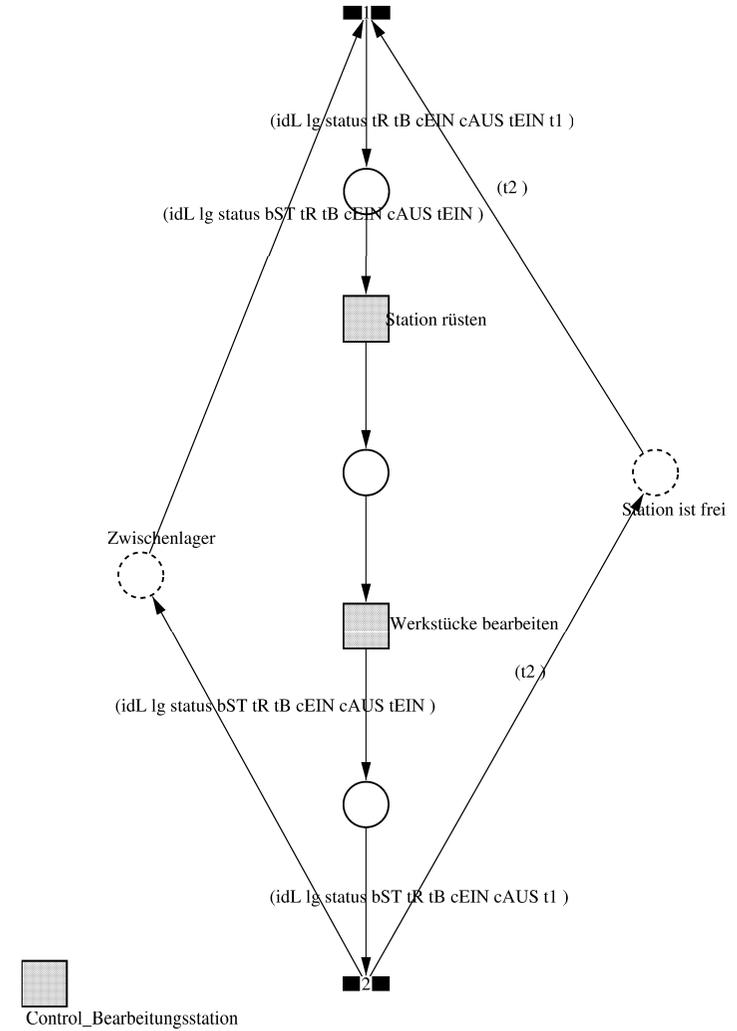
#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"
anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station =#manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station =#manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

```
#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel
```

====> Action Code <====

```
t2:= CurrentTime.
tEIN:= (CurrentTime) value.
```

```
"Status Los umsetzen"
status:= #wartet.
```

```
"Status der Bearbeitungsstation in Excel schreiben"
```

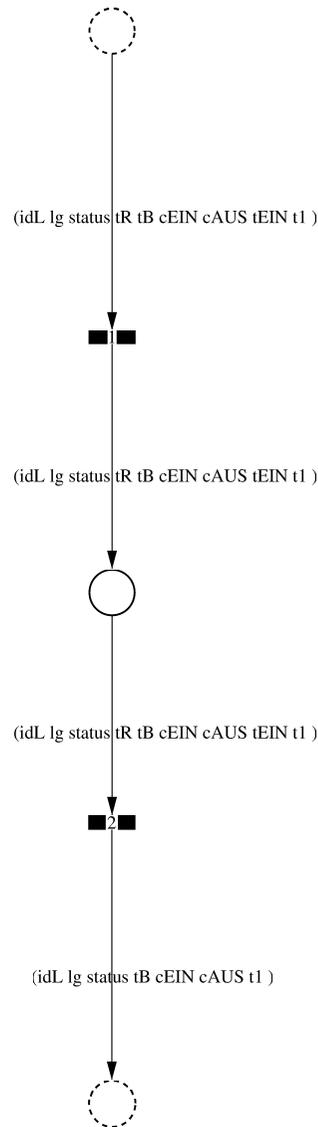
```
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:=(((rüsten*10) rounded)//10)
printString,',',(((rüsten*10)rounded)\10)*10)printString.
bearbeitenExcel:=(((bearbeiten*10) rounded)//10)
printString,',',(((bearbeiten*10)rounded)\10)*10)printString.
wartenExcel:=(((warten*10) rounded)//10)
printString,',',(((warten*10)rounded)\10)*10)printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.
```

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].
(station = #manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].
(station = #manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].
(station = #gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].
(station = #montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].
(station = #prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].
(station = #extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"
```

```
betriebszeit:= CurrentTime value.
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)
printString,',',(((betriebszeit*10)rounded)\10)*10)printString.
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

```
#tRNetto
#tR
#tRVar
#tRVarProzent
```

====> Action Code <====

```
"Rüstzeit und Schwankung auslesen und ausrechnen"
tRNetto:= tR removeFirst.
tRVarProzent:= tR removeFirst.
(tRVarProzent > 0) ifTrue:
[tRVar:= ((tRVarProzent/100) *tRNetto).
tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]
ifFalse:
[tR:= tRNetto].
```

2: Transition

====> Transition Variables <====

```
#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2
```

====> Delay Code <====

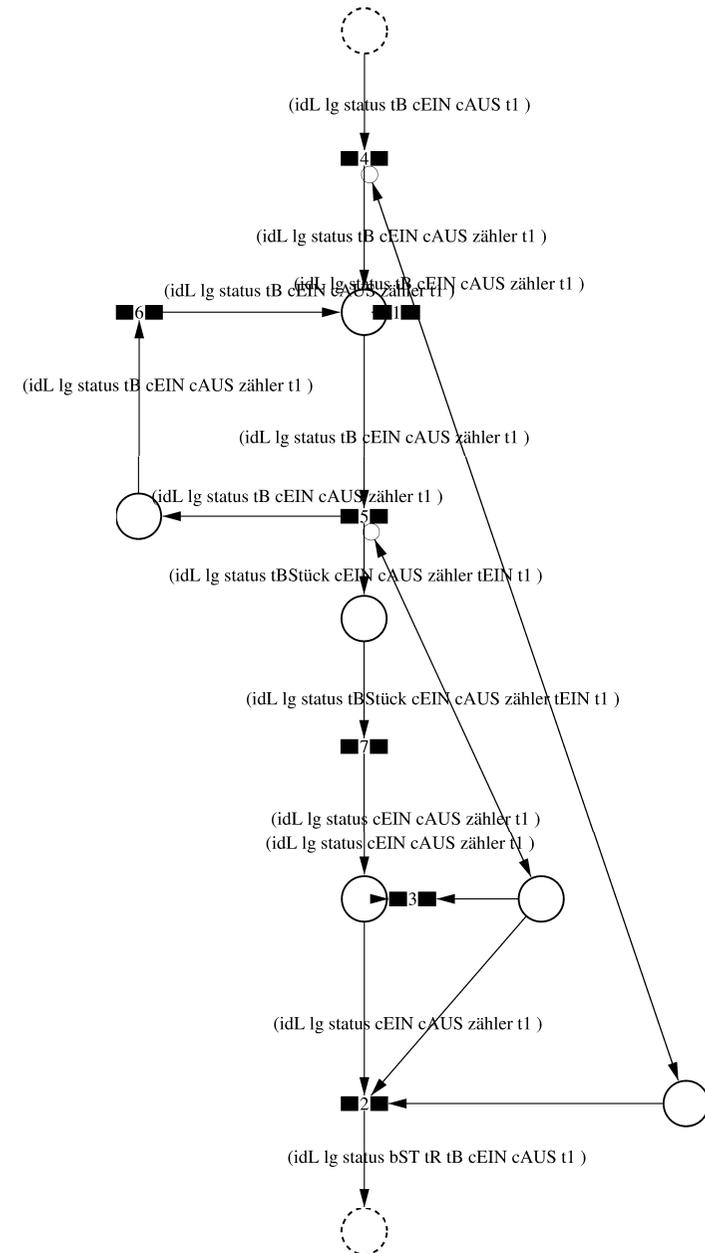
```
tR
```

====> Action Code <====

```
"Rüstzeit in cAUS schreiben"
tRüstAlt:= cAUS at: #tRüst.
tRüstHier:= ((CurrentTime) value) - tEIN.
tRüstNeu:= tRüstAlt + tRüstHier.
cAUS at: #tRüst put: tRüstNeu.
```

```
"#rüsten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
rüstenAlt:= (self at: #rüsten) value.
rüstenNeu:= (t2-tEIN) +rüstenAlt.
(self at: #rüsten) value: rüstenNeu.
```

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
ifTrue:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
ifFalse:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"

zähler:= 1

5: Transition

====> Transition Variables <====

#zähler

#tBNetto

#tBVar

#tBStück

#tBVarProzent

#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"

tBNetto:= tB at: 1.

tBVarProzent:= tB at: 2.

(tBVarProzent > 0) ifTrue:

[tBVar:= ((tBVarProzent/100) *tBNetto).

tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]

iffalse:

[tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"

status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"

zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt

#tBearbHier

#tBearbNeu

#t2

#bearbeitenAlt

#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"

tBearbAlt:= cAUS at: #tBearb.

tBearbHier:= ((CurrentTime) value) - tEIN.

tBearbNeu:= tBearbAlt + tBearbHier.

cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"

t2:= (CurrentTime) value.

bearbeitenAlt:= (self at: #bearbeiten) value.

bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.

(self at: #bearbeiten) value: bearbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.
Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

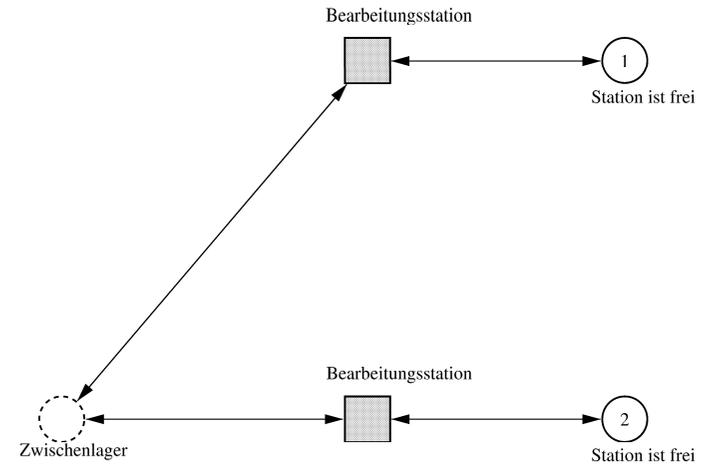
#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariabler auslesen, aktualisieren und in Excel schreiben"
anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station =#manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Montage



====> Development Comment <====

Modul: Kategorie Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Kategorie Bearbeitungsstation

Aufgabe: Dieses Modul "Kategorie Bearbeitungsstation" beinhaltet alle Bearbeitungsstationen EINES Typs (z.B. alle CNC-Drehmaschinen). Der Typ der Bearbeitungsstationen wird über die Modulvariable #maschine definiert. Die Anzahl der Bearbeitungsstationen in diesem Modul wird über die eingefügte Anzahl der Module "Bearbeitungsstation" eine Ebene tiefer definiert.

Anzahl der Eingangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Anzahl der Ausgangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Modulvariable Vorbesetzung (Initial) Beschreibung

#anzahl 0 Anzahl der Bearbeitungsstationen in diesem Modul.
#bearbeiten 0 Summiert die Bearbeitungszeiten der Bearbeitungsstationen während der Simulation auf.
#rüsten 0 Summiert die Rüstzeiten der Bearbeitungsstationen während der Simulation auf.
#station #eintragen Definition des Typs der Bearbeitungsstationen.
#warten 0 Summiert die Wartezeiten der Bearbeitungsstationen während der Simulation auf.

====> Net resp. Module Variables <====

#anzahl
#bearbeiten
#rüsten
#station

#warten

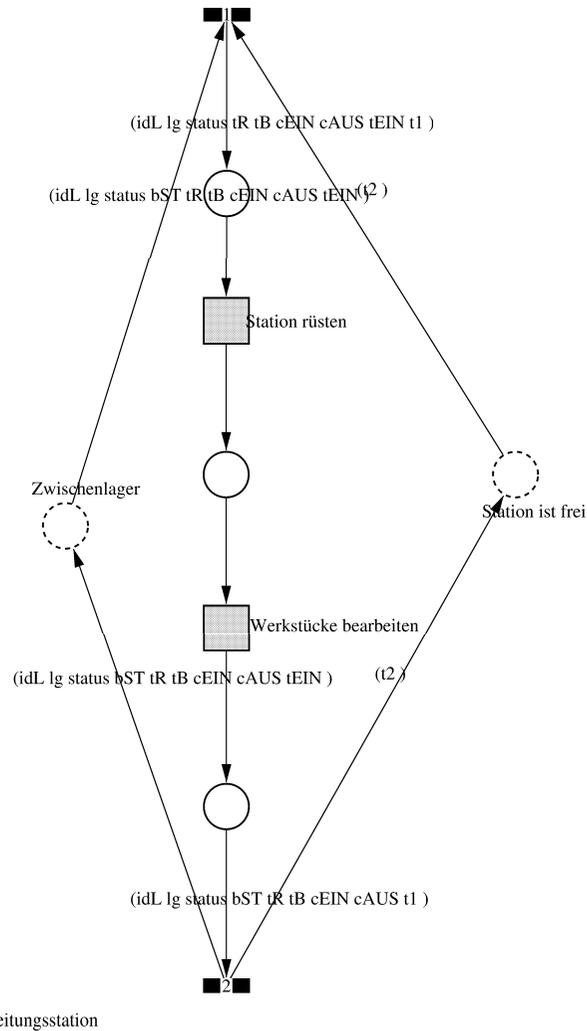
1: Place 'Station ist frei'

====> Initial Tokens <====
(CurrentTime)

2: Place 'Station ist frei'

====> Initial Tokens <====
(CurrentTime)

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	
Zwischenlager tEIN	Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.	
Station ist frei t2	Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.	

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel

====> Action Code <====

t2:= CurrentTime.
tEIN:= (CurrentTime) value.

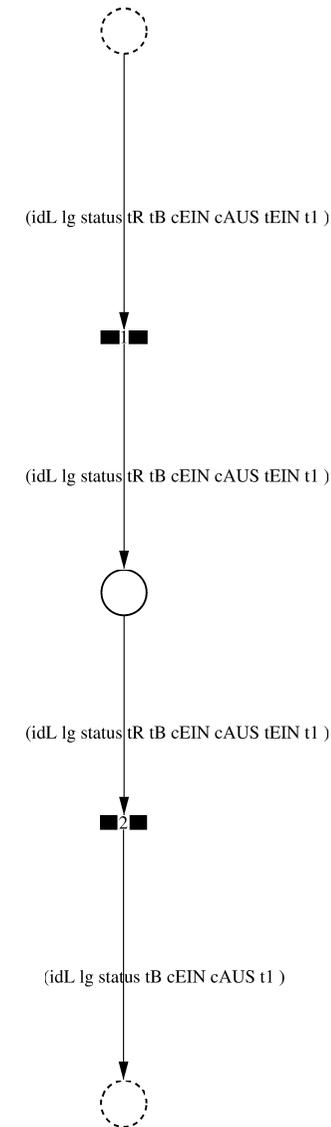
"Status Los umsetzen"
status:= #wartet.

"Status der Bearbeitungsstation in Excel schreiben"
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:= (((rüsten*10) rounded)//10)
printString, ', ', (((rüsten*10) rounded)\10)*10) printString.
bearbeitenExcel:= (((bearbeiten*10) rounded)//10)
printString, ', ', (((bearbeiten*10) rounded)\10)*10) printString.
wartenExcel:= (((warten*10) rounded)//10)
printString, ', ', (((warten*10) rounded)\10)*10) printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].  
(station =#manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].  
(station =#manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].  
(station =#montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"  
betriebszeit:= CurrentTime value.  
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)  
printString, ', ',(((betriebszeit*10)rounded)\10)*10)printString.  
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"

tRNetto:= tR removeFirst.

tRVarProzent:= tR removeFirst.

(tRVarProzent > 0) ifTrue:

[tRVar:= ((tRVarProzent/100) *tRNetto).

tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]

ifFalse:

[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"

tRüstAlt:= cAUS at: #tRüst.

tRüstHier:= ((CurrentTime) value) - tEIN.

tRüstNeu:= tRüstAlt + tRüstHier.

cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"

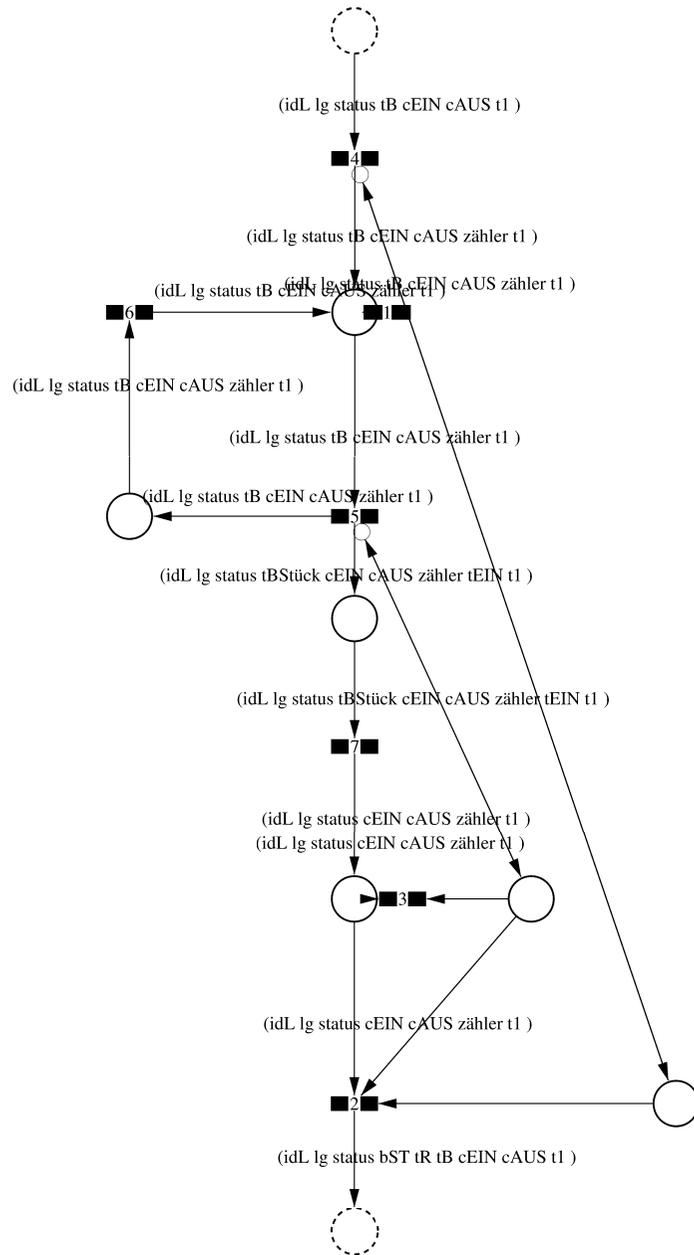
t2:= (CurrentTime) value.

rüstenAlt:= (self at: #rüsten) value.

rüstenNeu:= (t2-tEIN) +rüstenAlt.

(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- $status$ Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- $cEIN$ Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- $cAUS$ Collection, welche die Auswerte-Daten des Loses speichert.
- $t1$ Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- $status$ Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- $cEIN$ Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- $cAUS$ Collection, welche die Auswerte-Daten des Loses speichert.
- $t1$ Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
if True:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
if False:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"
zähler:= 1

5: Transition

====> Transition Variables <====

#zähler
#tBNetto
#tBVar
#tBStück
#tBVarProzent
#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"
tBNetto:= tB at: 1.
tBVarProzent:= tB at: 2.
(tBVarProzent > 0) if True:
 [tBVar:= ((tBVarProzent/100) *tBNetto).
 tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]
if False:
 [tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"
status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"
zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt
#tBearbHier
#tBearbNeu
#t2
#bearbeitenAlt
#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"
tBearbAlt:= cAUS at: #tBearb.
tBearbHier:= ((CurrentTime) value) - tEIN.
tBearbNeu:= tBearbAlt + tBearbHier.
cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
bearbeitenAlt:= (self at: #bearbeiten) value.
bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.
(self at: #bearbeiten) value: arbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.

Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

#anzahlAlt
#anzahlNeu
#station

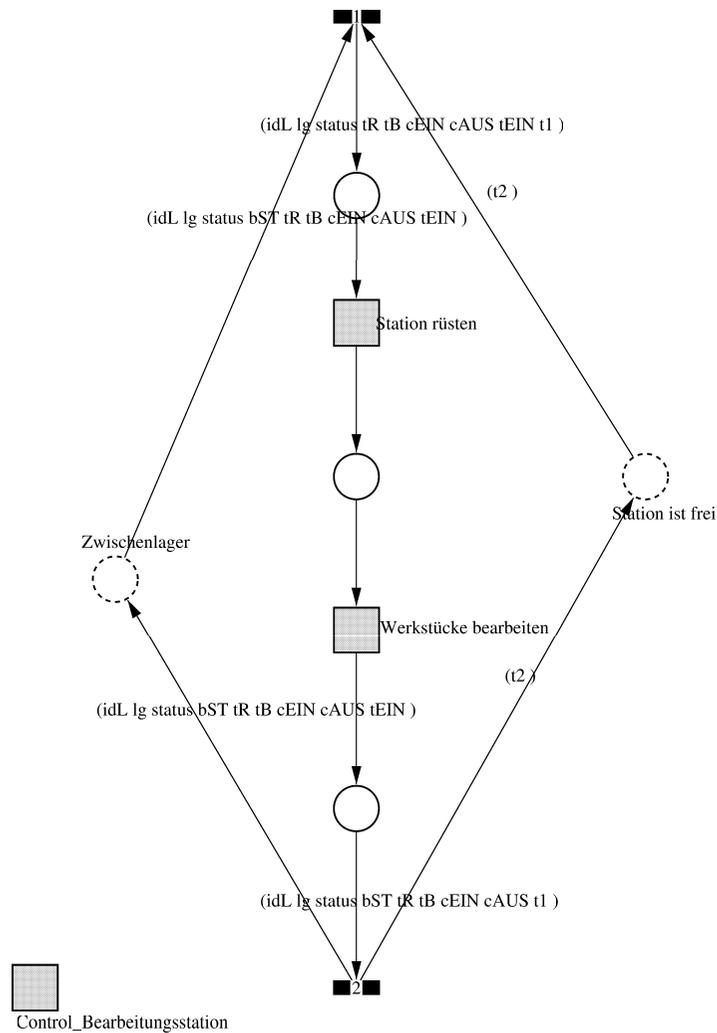
====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"

anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station = #manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	
Zwischenlager tEIN	Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.	
Station ist frei t2	Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.	

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel

====> Action Code <====

t2:= CurrentTime.
tEIN:= (CurrentTime) value.

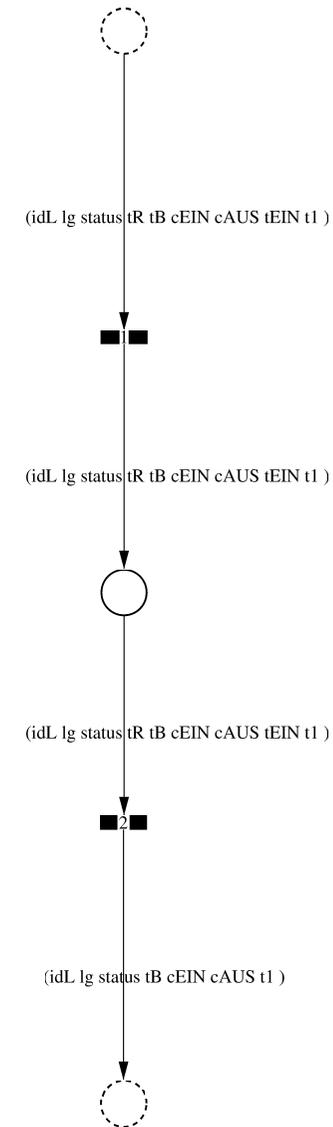
"Status Los umsetzen"
status:= #wartet.

"Status der Bearbeitungsstation in Excel schreiben"
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:= (((rüsten*10) rounded)//10)
printString, ', ', (((rüsten*10) rounded)\10)*10) printString.
bearbeitenExcel:= (((bearbeiten*10) rounded)//10)
printString, ', ', (((bearbeiten*10) rounded)\10)*10) printString.
wartenExcel:= (((warten*10) rounded)//10)
printString, ', ', (((warten*10) rounded)\10)*10) printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].  
(station =#manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].  
(station =#manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].  
(station =#montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"  
betriebszeit:= CurrentTime value.  
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)  
printString, ', ',(((betriebszeit*10)rounded)\10)*10)printString.  
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"

tRNetto:= tR removeFirst.

tRVarProzent:= tR removeFirst.

(tRVarProzent > 0) ifTrue:

[tRVar:= ((tRVarProzent/100) *tRNetto).

tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]

ifFalse:

[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"

tRüstAlt:= cAUS at: #tRüst.

tRüstHier:= ((CurrentTime) value) - tEIN.

tRüstNeu:= tRüstAlt + tRüstHier.

cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"

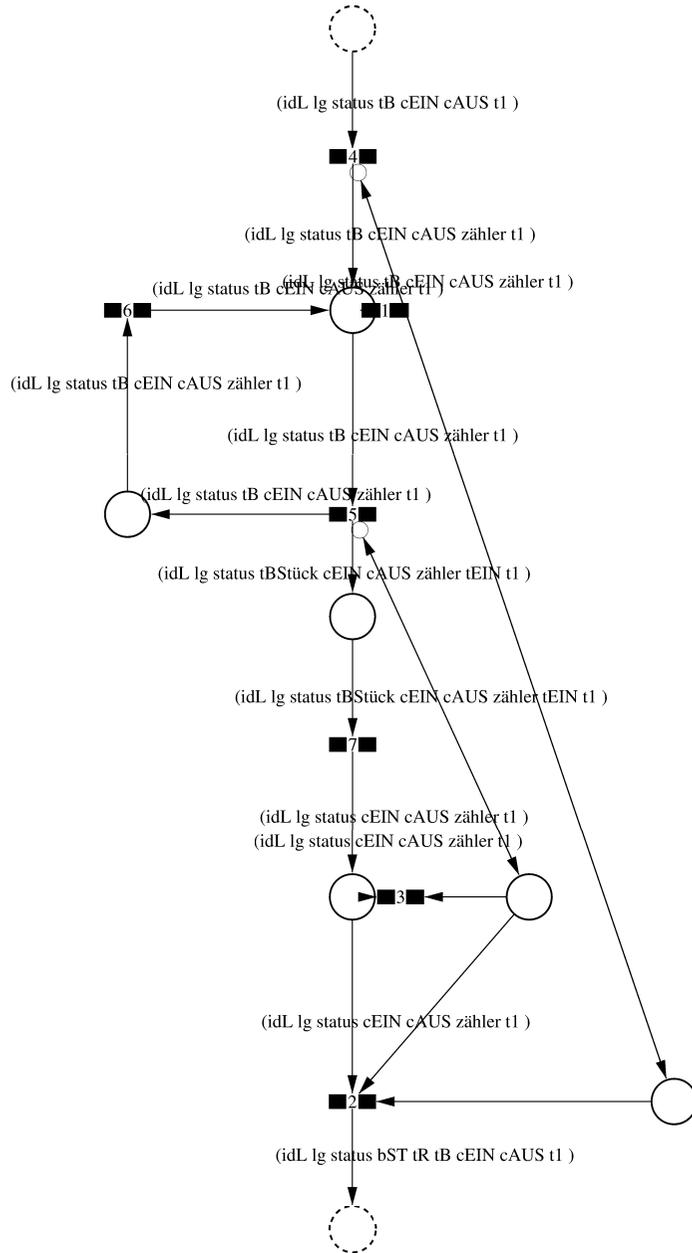
t2:= (CurrentTime) value.

rüstenAlt:= (self at: #rüsten) value.

rüstenNeu:= (t2-tEIN) +rüstenAlt.

(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
if True:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
if False:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"
zähler:= 1

5: Transition

====> Transition Variables <====

#zähler
#tBNetto
#tBVar
#tBStück
#tBVarProzent
#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"
tBNetto:= tB at: 1.
tBVarProzent:= tB at: 2.
(tBVarProzent > 0) if True:
 [tBVar:= ((tBVarProzent/100) *tBNetto).
 tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]
if False:
 [tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"
status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"
zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt
#tBearbHier
#tBearbNeu
#t2
#bearbeitenAlt
#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"
tBearbAlt:= cAUS at: #tBearb.
tBearbHier:= ((CurrentTime) value) - tEIN.
tBearbNeu:= tBearbAlt + tBearbHier.
cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
bearbeitenAlt:= (self at: #bearbeiten) value.
bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.
(self at: #bearbeiten) value: arbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.

Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"

anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

station:= (self at: #station) value.

Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.

(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu) printString.].

(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu) printString.].

(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu) printString.].

(station = #manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu) printString.].

(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu) printString.].

(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu) printString.].

(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu) printString.].

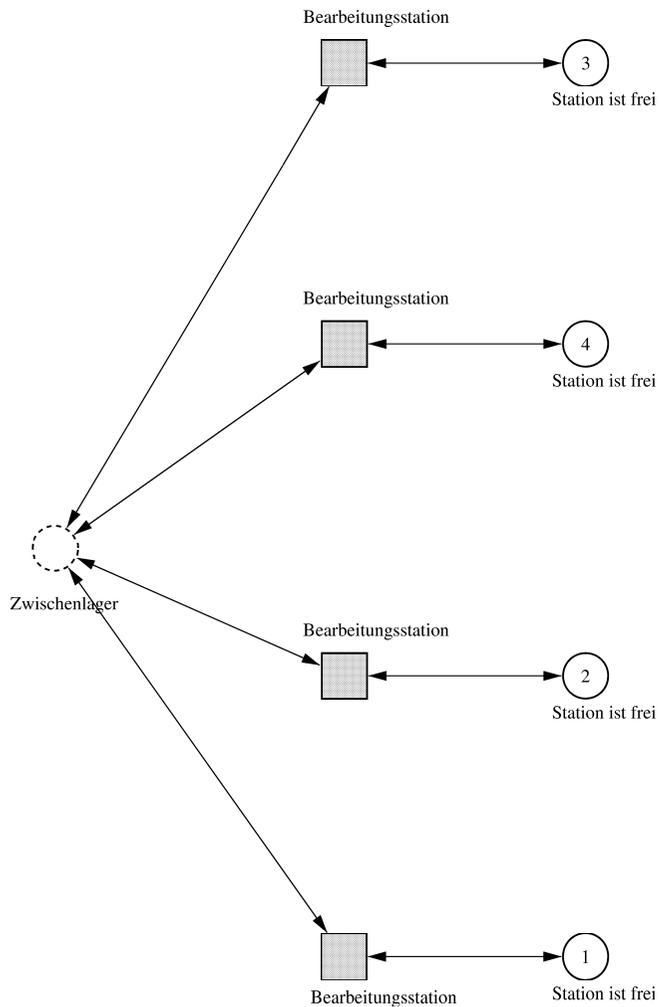
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu) printString.].

(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu) printString.].

(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu) printString.].

(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu) printString.].

Module: CNC-Drehmaschinen



====> Development Comment <====

Modul: Kategorie Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Kategorie Bearbeitungsstation

Aufgabe: Dieses Modul "Kategorie Bearbeitungsstation" beinhaltet alle Bearbeitungsstationen EINES Typs (z.B. alle CNC-Drehmaschinen). Der Typ der Bearbeitungsstationen wird über die Modulvariable #maschine definiert. Die Anzahl der Bearbeitungsstationen in diesem Modul wird über die eingefügte Anzahl der Module "Bearbeitungsstation" eine Ebene tiefer definiert.

Anzahl der Eingangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Anzahl der Ausgangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Modulvariable Vorbesetzung (Initial) Beschreibung

#anzahl 0 Anzahl der Bearbeitungsstationen in diesem Modul.
#bearbeiten 0 Summiert die Bearbeitungszeiten der Bearbeitungsstationen während der Simulation auf.
#rüsten 0 Summiert die Rüstzeiten der Bearbeitungsstationen während der Simulation auf.
#station #eintragen Definition des Typs der Bearbeitungsstationen.
#warten 0 Summiert die Wartezeiten der Bearbeitungsstationen während der Simulation auf.

====> Net resp. Module Variables <====

#anzahl
#bearbeiten
#rüsten
#station

#warten

1: Place 'Station ist frei'

====> Initial Tokens <=====
(CurrentTime)

2: Place 'Station ist frei'

====> Initial Tokens <=====
(CurrentTime)

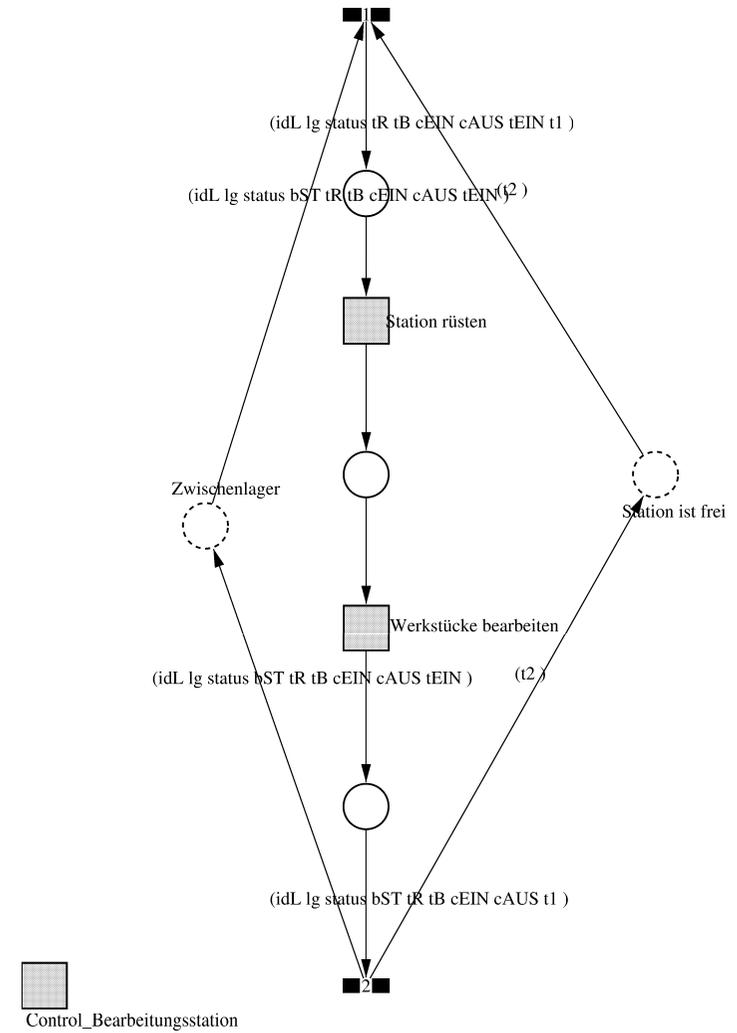
3: Place 'Station ist frei'

====> Initial Tokens <=====
(CurrentTime)

4: Place 'Station ist frei'

====> Initial Tokens <=====
(CurrentTime)

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

```
#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel
```

====> Action Code <====

```
t2:= CurrentTime.
tEIN:= (CurrentTime) value.
```

```
"Status Los umsetzen"
status:= #wartet.
```

```
"Status der Bearbeitungsstation in Excel schreiben"
```

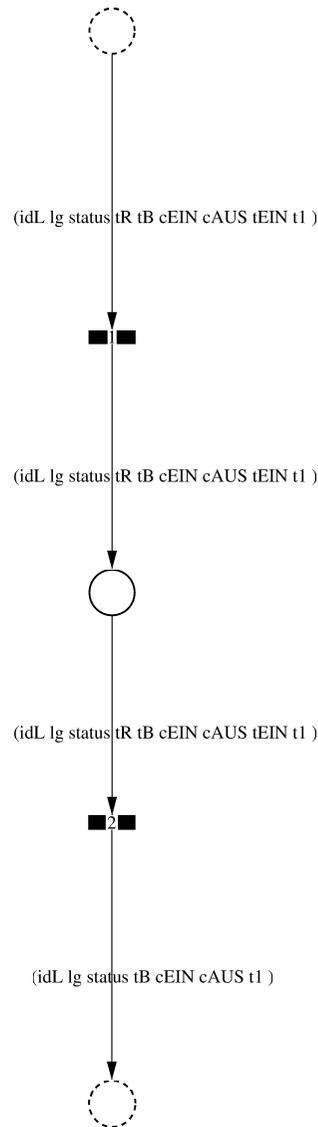
```
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:=(((rüsten*10) rounded)//10)
printString,',',(((rüsten*10)rounded)\10)*10)printString.
bearbeitenExcel:=(((bearbeiten*10) rounded)//10)
printString,',',(((bearbeiten*10)rounded)\10)*10)printString.
wartenExcel:=(((warten*10) rounded)//10)
printString,',',(((warten*10)rounded)\10)*10)printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.
```

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].
(station = #manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].
(station = #manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].
(station = #gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].
(station = #montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].
(station = #prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].
(station = #extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"
```

```
betriebszeit:= CurrentTime value.
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)
printString,',',(((betriebszeit*10)rounded)\10)*10)printString.
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"
tRNetto:= tR removeFirst.
tRVarProzent:= tR removeFirst.
(tRVarProzent > 0) ifTrue:
[tRVar:= ((tRVarProzent/100) *tRNetto).
tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]
ifFalse:
[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

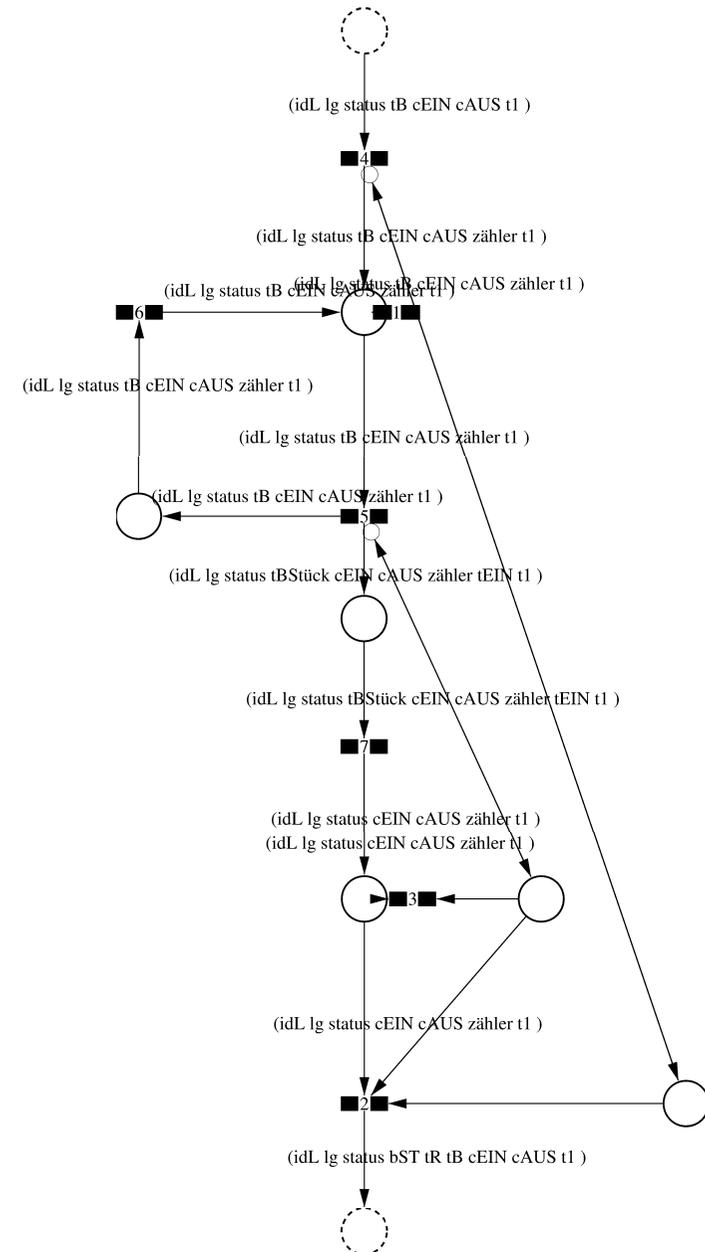
tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"
tRüstAlt:= cAUS at: #tRüst.
tRüstHier:= ((CurrentTime) value) - tEIN.
tRüstNeu:= tRüstAlt + tRüstHier.
cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
rüstenAlt:= (self at: #rüsten) value.
rüstenNeu:= (t2-tEIN) +rüstenAlt.
(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
ifTrue:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
ifFalse:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"

zähler:= 1

5: Transition

====> Transition Variables <====

#zähler

#tBNetto

#tBVar

#tBStück

#tBVarProzent

#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"

tBNetto:= tB at: 1.

tBVarProzent:= tB at: 2.

(tBVarProzent > 0) ifTrue:

[tBVar:= ((tBVarProzent/100) *tBNetto).

tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]

iffalse:

[tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"

status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"

zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt

#tBearbHier

#tBearbNeu

#t2

#bearbeitenAlt

#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"

tBearbAlt:= cAUS at: #tBearb.

tBearbHier:= ((CurrentTime) value) - tEIN.

tBearbNeu:= tBearbAlt + tBearbHier.

cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"

t2:= (CurrentTime) value.

bearbeitenAlt:= (self at: #bearbeiten) value.

bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.

(self at: #bearbeiten) value: arbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



2

====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.
Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

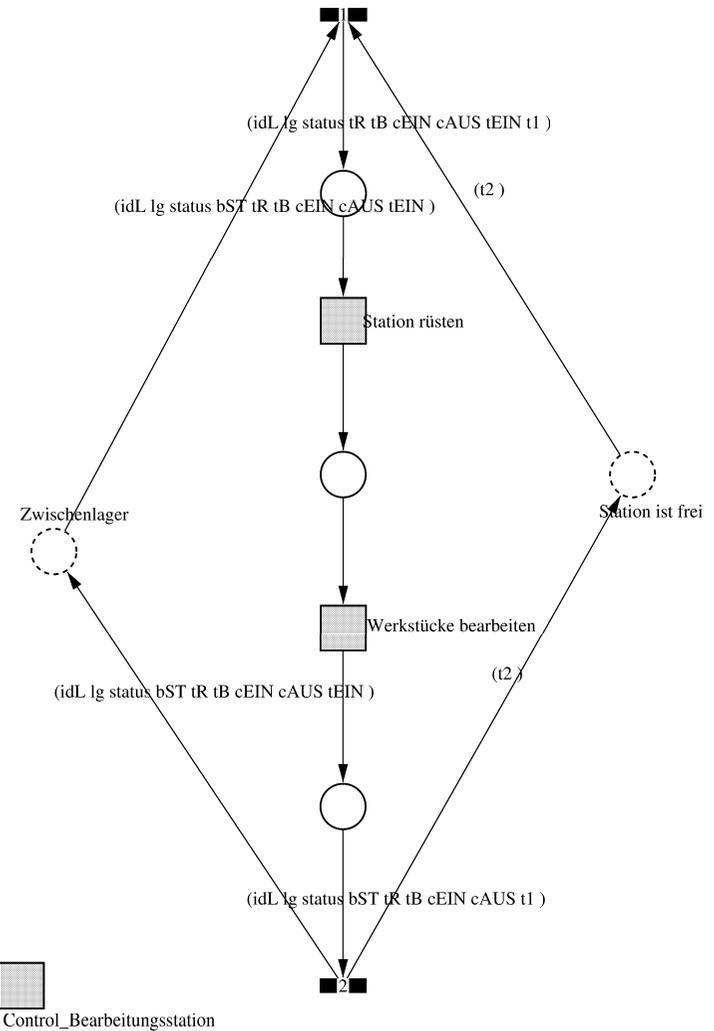
#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariabler auslesen, aktualisieren und in Excel schreiben"
anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station =#manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station =#manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

```
#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel
```

```
====> Action Code <====
```

```
t2:= CurrentTime.
tEIN:= (CurrentTime) value.
```

```
"Status Los umsetzen"
status:= #wartet.
```

```
"Status der Bearbeitungsstation in Excel schreiben"
```

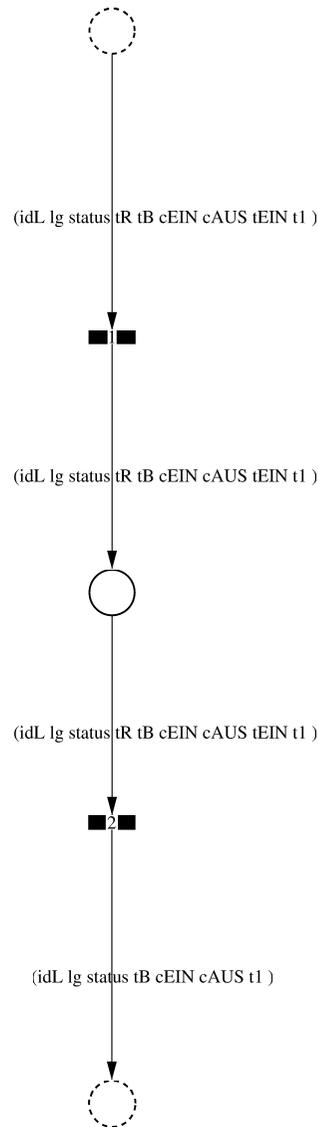
```
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:=(((rüsten*10) rounded)//10)
printString,',',(((rüsten*10)rounded)\10)*10)printString.
bearbeitenExcel:=(((bearbeiten*10) rounded)//10)
printString,',',(((bearbeiten*10)rounded)\10)*10)printString.
wartenExcel:=(((warten*10) rounded)//10)
printString,',',(((warten*10)rounded)\10)*10)printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.
```

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].
(station = #manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].
(station = #manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].
(station = #gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].
(station = #montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].
(station = #prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].
(station = #extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"
```

```
betriebszeit:= CurrentTime value.
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)
printString,',',(((betriebszeit*10)rounded)\10)*10)printString.
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"
tRNetto:= tR removeFirst.
tRVarProzent:= tR removeFirst.
(tRVarProzent > 0) ifTrue:
[tRVar:= ((tRVarProzent/100) *tRNetto).
tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]
ifFalse:
[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

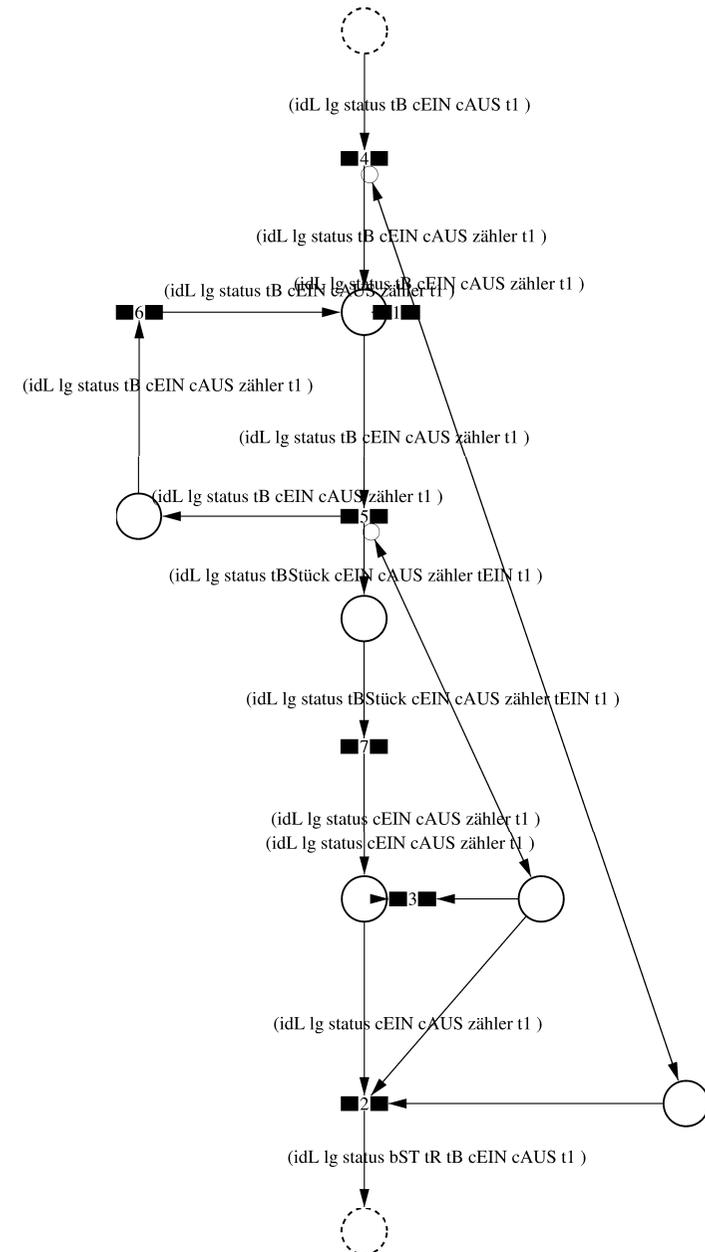
tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"
tRüstAlt:= cAUS at: #tRüst.
tRüstHier:= ((CurrentTime) value) - tEIN.
tRüstNeu:= tRüstAlt + tRüstHier.
cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
rüstenAlt:= (self at: #rüsten) value.
rüstenNeu:= (t2-tEIN) +rüstenAlt.
(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
ifTrue:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
ifFalse:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"

zähler:= 1

5: Transition

====> Transition Variables <====

#zähler

#tBNetto

#tBVar

#tBStück

#tBVarProzent

#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"

tBNetto:= tB at: 1.

tBVarProzent:= tB at: 2.

(tBVarProzent > 0) ifTrue:

[tBVar:= ((tBVarProzent/100) *tBNetto).

tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]

iffalse:

[tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"

status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"

zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt

#tBearbHier

#tBearbNeu

#t2

#bearbeitenAlt

#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"

tBearbAlt:= cAUS at: #tBearb.

tBearbHier:= ((CurrentTime) value) - tEIN.

tBearbNeu:= tBearbAlt + tBearbHier.

cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"

t2:= (CurrentTime) value.

bearbeitenAlt:= (self at: #bearbeiten) value.

bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.

(self at: #bearbeiten) value: bearbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



2

====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.
Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

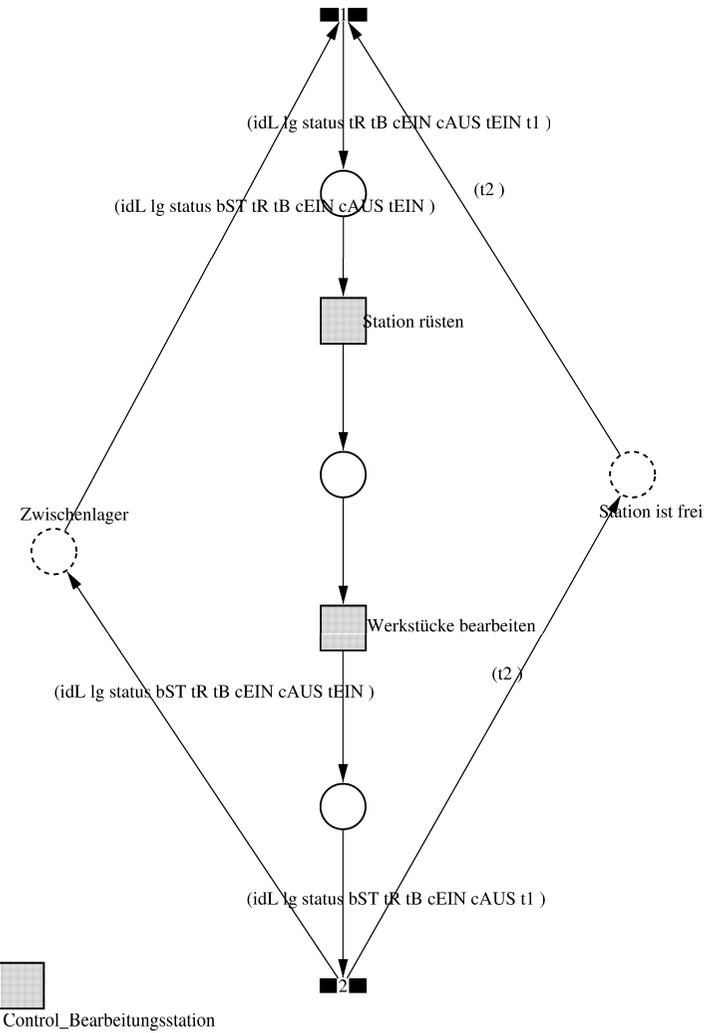
#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariabler auslesen, aktualisieren und in Excel schreiben"
anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station =#manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station =#manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

```
#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel
```

====> Action Code <====

```
t2:= CurrentTime.
tEIN:= (CurrentTime) value.
```

```
"Status Los umsetzen"
status:= #wartet.
```

```
"Status der Bearbeitungsstation in Excel schreiben"
```

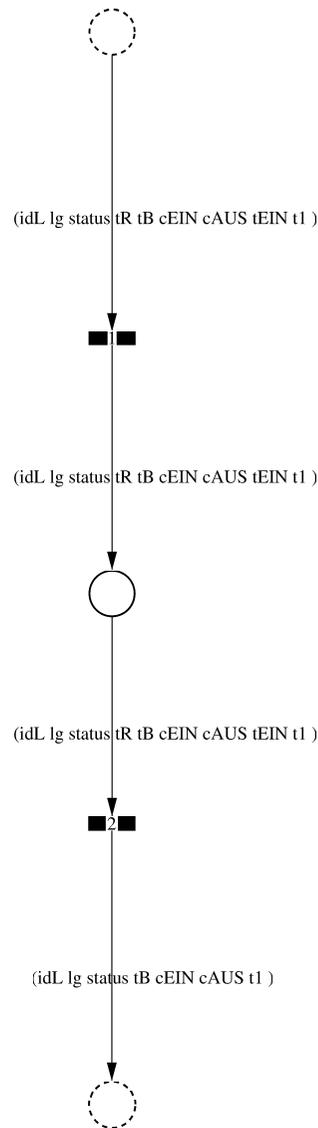
```
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:=(((rüsten*10) rounded)//10)
printString,',',(((rüsten*10)rounded)\10)*10)printString.
bearbeitenExcel:=(((bearbeiten*10) rounded)//10)
printString,',',(((bearbeiten*10)rounded)\10)*10)printString.
wartenExcel:=(((warten*10) rounded)//10)
printString,',',(((warten*10)rounded)\10)*10)printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.
```

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].
(station = #manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].
(station = #manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].
(station = #gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].
(station = #montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].
(station = #prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].
(station = #extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"
```

```
betriebszeit:= CurrentTime value.
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)
printString,',',(((betriebszeit*10)rounded)\10)*10)printString.
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"

tRNetto:= tR removeFirst.

tRVarProzent:= tR removeFirst.

(tRVarProzent > 0) ifTrue:

[tRVar:= ((tRVarProzent/100) *tRNetto).

tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]

ifFalse:

[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt

#tRüstHier

#tRüstNeu

#rüstenAlt

#rüstenNeu

#t2

====> Delay Code <====

tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"

tRüstAlt:= cAUS at: #tRüst.

tRüstHier:= ((CurrentTime) value) - tEIN.

tRüstNeu:= tRüstAlt + tRüstHier.

cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"

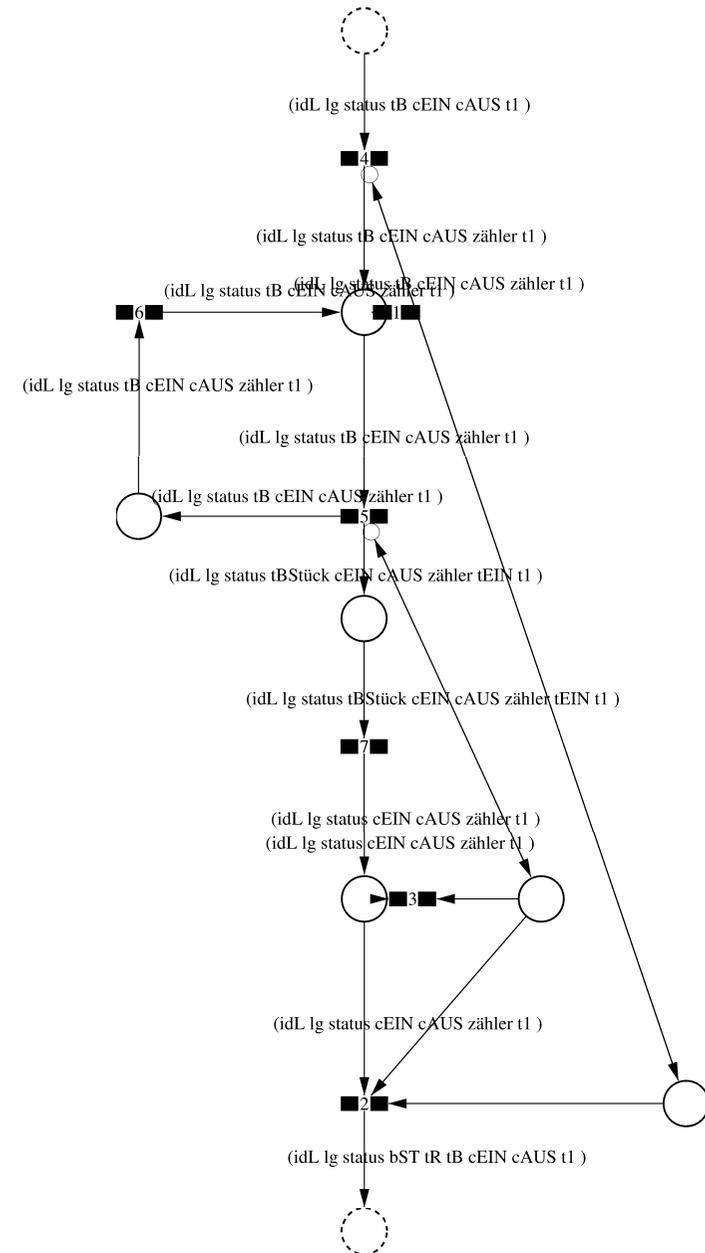
t2:= (CurrentTime) value.

rüstenAlt:= (self at: #rüsten) value.

rüstenNeu:= (t2-tEIN) +rüstenAlt.

(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
 - lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
 - status Zeigt den aktuellen Status des Loses in der Simulation an.
 - tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
 - cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
 - cAUS Collection, welche die Auswerte-Daten des Loses speichert.
 - t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
 - lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
 - status Zeigt den aktuellen Status des Loses in der Simulation an.
 - bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
 - tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
 - tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
 - cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
 - cAUS Collection, welche die Auswerte-Daten des Loses speichert.
 - t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
ifTrue:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
ifFalse:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"

zähler:= 1

5: Transition

====> Transition Variables <====

#zähler

#tBNetto

#tBVar

#tBStück

#tBVarProzent

#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"

tBNetto:= tB at: 1.

tBVarProzent:= tB at: 2.

(tBVarProzent > 0) ifTrue:

[tBVar:= ((tBVarProzent/100) *tBNetto).

tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]

iffalse:

[tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"

status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"

zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt

#tBearbHier

#tBearbNeu

#t2

#bearbeitenAlt

#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"

tBearbAlt:= cAUS at: #tBearb.

tBearbHier:= ((CurrentTime) value) - tEIN.

tBearbNeu:= tBearbAlt + tBearbHier.

cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"

t2:= (CurrentTime) value.

bearbeitenAlt:= (self at: #bearbeiten) value.

bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.

(self at: #bearbeiten) value: bearbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



2

====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.
Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

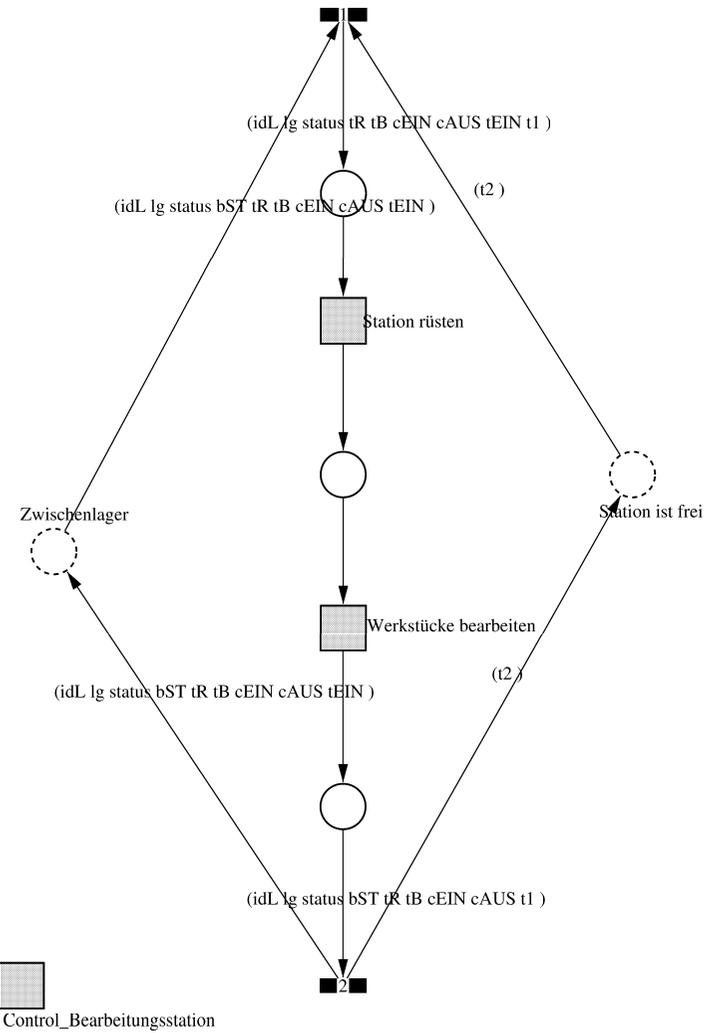
#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"
anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station =#manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station =#manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

```
#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel
```

====> Action Code <====

```
t2:= CurrentTime.
tEIN:= (CurrentTime) value.
```

```
"Status Los umsetzen"
status:= #wartet.
```

```
"Status der Bearbeitungsstation in Excel schreiben"
```

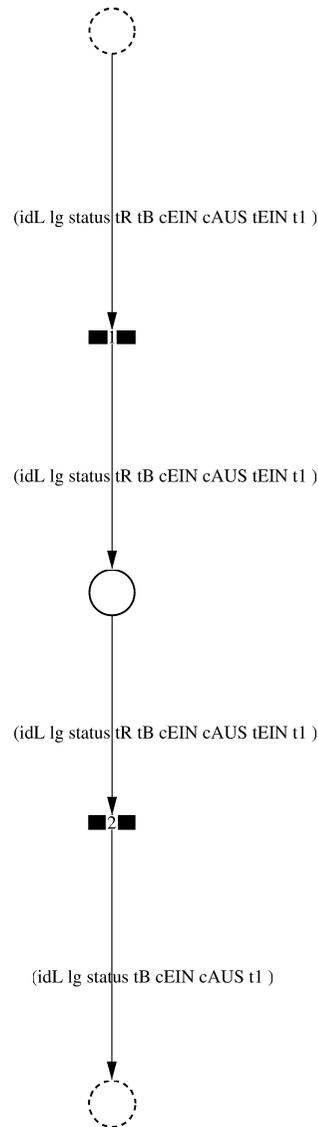
```
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:=(((rüsten*10) rounded)//10)
printString,',',(((rüsten*10)rounded)\10)*10)printString.
bearbeitenExcel:=(((bearbeiten*10) rounded)//10)
printString,',',(((bearbeiten*10)rounded)\10)*10)printString.
wartenExcel:=(((warten*10) rounded)//10)
printString,',',(((warten*10)rounded)\10)*10)printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.
```

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].
(station = #manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].
(station = #manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].
(station = #gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].
(station = #montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].
(station = #prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].
(station = #extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"
```

```
betriebszeit:= CurrentTime value.
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)
printString,',',(((betriebszeit*10)rounded)\10)*10)printString.
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- | Bezeichner | Attribute | Beschreibung |
|------------|--|--------------|
| -- idL | Eindeutige Nummer des Loses, in Excel festgelegt. | |
| -- lg | Anzahl der Werkstücke in dem Los, in Excel festgelegt. | |
| -- status | Zeigt den aktuellen Status des Loses in der Simulation an. | |
| -- tR | Gibt die Rüstzeit des Loses für die Bearbeitungsstation an. | |
| -- tB | Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an. | |
| -- cEIN | Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert. | |
| -- cAUS | Collection, welche die Auswerte-Daten des Loses speichert. | |
| -- tEIN | Wird zur Auswertung der Wartezeit im Zwischenlager benötigt. | |
| -- t1 | Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt. | |

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- | Bezeichner | Attribute | Beschreibung |
|------------|--|--------------|
| -- idL | Eindeutige Nummer des Loses, in Excel festgelegt. | |
| -- lg | Anzahl der Werkstücke in dem Los, in Excel festgelegt. | |
| -- status | Zeigt den aktuellen Status des Loses in der Simulation an. | |
| -- tB | Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an. | |
| -- cEIN | Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert. | |
| -- cAUS | Collection, welche die Auswerte-Daten des Loses speichert. | |
| -- t1 | Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt. | |

1: Transition

====> Transition Variables <====

#tRNetto
 #tR
 #tRVar
 #tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"
 tRNetto:= tR removeFirst.
 tRVarProzent:= tR removeFirst.
 (tRVarProzent > 0) ifTrue:
 [tRVar:= ((tRVarProzent/100) *tRNetto).
 tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]
 ifFalse:
 [tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
 #tRüstHier
 #tRüstNeu
 #rüstenAlt
 #rüstenNeu
 #t2

====> Delay Code <====

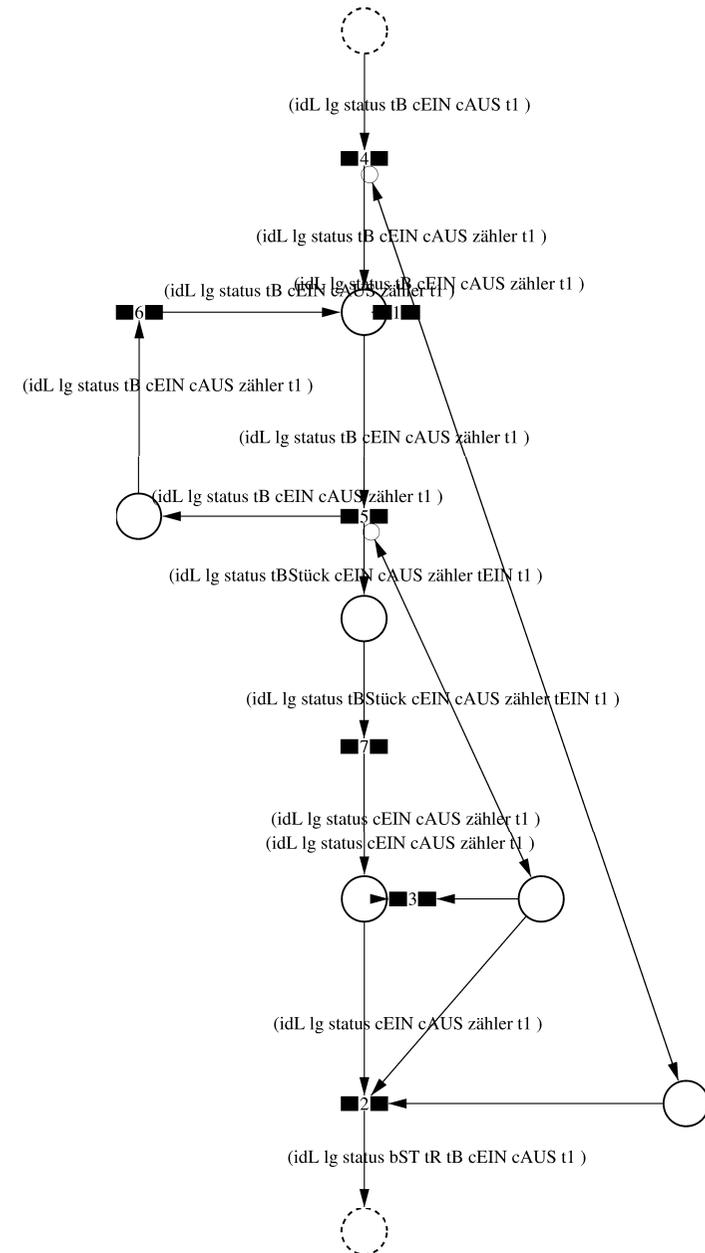
tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"
 tRüstAlt:= cAUS at: #tRüst.
 tRüstHier:= ((CurrentTime) value) - tEIN.
 tRüstNeu:= tRüstAlt + tRüstHier.
 cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"
 t2:= (CurrentTime) value.
 rüstenAlt:= (self at: #rüsten) value.
 rüstenNeu:= (t2-tEIN) +rüstenAlt.
 (self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- | Bezeichner | Attribute | Beschreibung |
|------------|--|--------------|
| -- idL | Eindeutige Nummer des Loses, in Excel festgelegt. | |
| -- lg | Anzahl der Werkstücke in dem Los, in Excel festgelegt. | |
| -- status | Zeigt den aktuellen Status des Loses in der Simulation an. | |
| -- tB | Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an. | |
| -- cEIN | Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert. | |
| -- cAUS | Collection, welche die Auswerte-Daten des Loses speichert. | |
| -- t1 | Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt. | |

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- | Bezeichner | Attribute | Beschreibung |
|------------|--|--------------|
| -- idL | Eindeutige Nummer des Loses, in Excel festgelegt. | |
| -- lg | Anzahl der Werkstücke in dem Los, in Excel festgelegt. | |
| -- status | Zeigt den aktuellen Status des Loses in der Simulation an. | |
| -- bST | Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird. | |
| -- tR | Gibt die Rüstzeit des Loses für die Bearbeitungsstation an. | |
| -- tB | Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an. | |
| -- cEIN | Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert. | |
| -- cAUS | Collection, welche die Auswerte-Daten des Loses speichert. | |
| -- t1 | Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt. | |

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
ifTrue:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
ifFalse:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"

zähler:= 1

5: Transition

====> Transition Variables <====

#zähler

#tBNetto

#tBVar

#tBStück

#tBVarProzent

#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"

tBNetto:= tB at: 1.

tBVarProzent:= tB at: 2.

(tBVarProzent > 0) ifTrue:

[tBVar:= ((tBVarProzent/100) *tBNetto).

tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]

iffalse:

[tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"

status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"

zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt

#tBearbHier

#tBearbNeu

#t2

#bearbeitenAlt

#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"

tBearbAlt:= cAUS at: #tBearb.

tBearbHier:= ((CurrentTime) value) - tEIN.

tBearbNeu:= tBearbAlt + tBearbHier.

cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"

t2:= (CurrentTime) value.

bearbeitenAlt:= (self at: #bearbeiten) value.

bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.

(self at: #bearbeiten) value: arbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.
Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

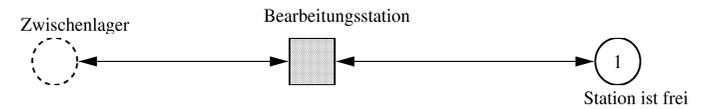
#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"
anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station =#manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: CNC-Dreh-Fräszentren



====> Development Comment <====

Modul: Kategorie Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Kategorie Bearbeitungsstation

Aufgabe: Dieses Modul "Kategorie Bearbeitungsstation" beinhaltet alle Bearbeitungsstationen EINES Typs (z.B. alle CNC-Drehmaschinen). Der Typ der Bearbeitungsstationen wird über die Modulvariable #maschine definiert. Die Anzahl der Bearbeitungsstationen in diesem Modul wird über die eingefügte Anzahl der Module "Bearbeitungsstation" eine Ebene tiefer definiert.

Anzahl der Eingangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Anzahl der Ausgangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Modulvariable Vorbesetzung (Initial) Beschreibung

#anzahl 0 Anzahl der Bearbeitungsstationen in diesem Modul.
#bearbeiten 0 Summiert die Bearbeitungszeiten der Bearbeitungsstationen während der Simulation auf.
#rüsten 0 Summiert die Rüstzeiten der Bearbeitungsstationen während der Simulation auf.
#station #eintragen Definition des Typs der Bearbeitungsstationen.
#warten 0 Summiert die Wartezeiten der Bearbeitungsstationen während der Simulation auf.

====> Net resp. Module Variables <====

#anzahl
#bearbeiten
#rüsten
#station

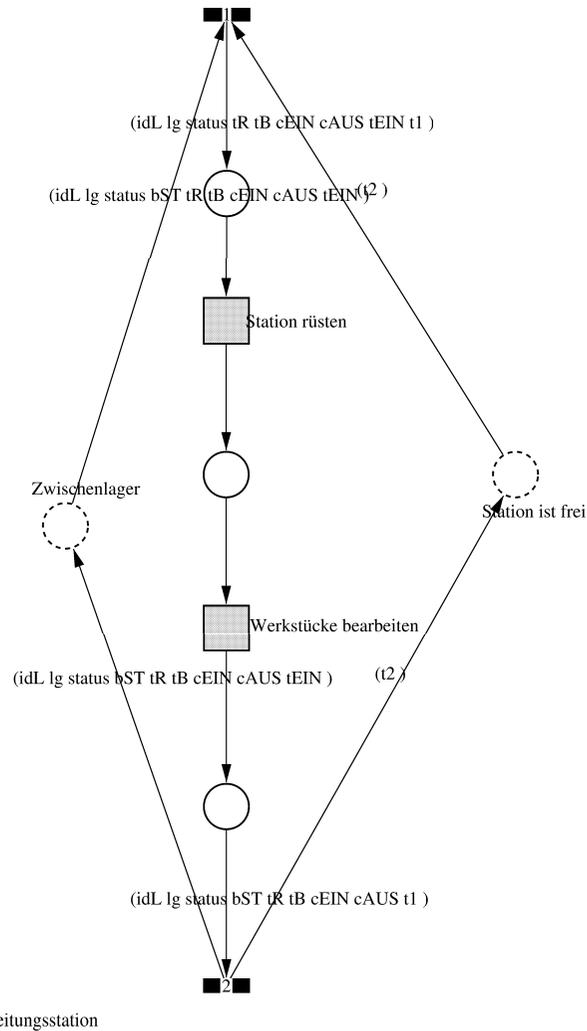
#warten

1: Place 'Station ist frei'

====> Initial Tokens <====

(CurrentTime)

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	
Zwischenlager tEIN	Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.	
Station ist frei t2	Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.	

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel

====> Action Code <====

t2:= CurrentTime.
tEIN:= (CurrentTime) value.

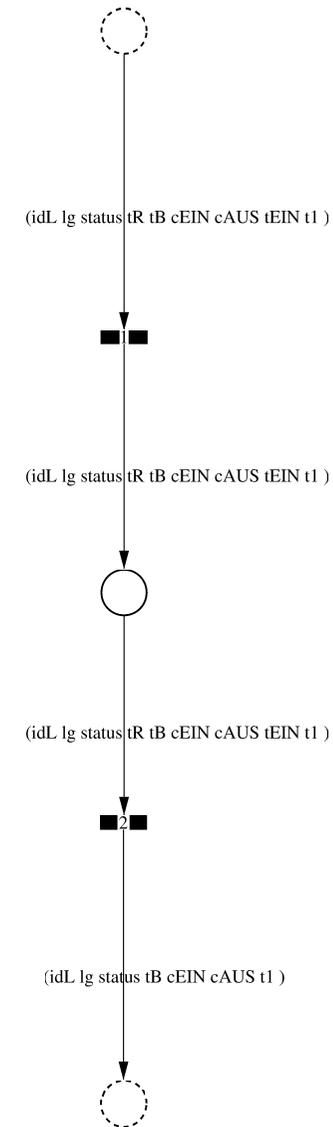
"Status Los umsetzen"
status:= #wartet.

"Status der Bearbeitungsstation in Excel schreiben"
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:= (((rüsten*10) rounded)//10)
printString, ', ', (((rüsten*10) rounded)\10)*10) printString.
bearbeitenExcel:= (((bearbeiten*10) rounded)//10)
printString, ', ', (((bearbeiten*10) rounded)\10)*10) printString.
wartenExcel:= (((warten*10) rounded)//10)
printString, ', ', (((warten*10) rounded)\10)*10) printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].  
(station =#manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].  
(station =#manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].  
(station =#montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"  
betriebszeit:= CurrentTime value.  
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)  
printString, ', ',(((betriebszeit*10)rounded)\10)*10)printString.  
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"

tRNetto:= tR removeFirst.

tRVarProzent:= tR removeFirst.

(tRVarProzent > 0) ifTrue:

[tRVar:= ((tRVarProzent/100) *tRNetto).

tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]

ifFalse:

[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"

tRüstAlt:= cAUS at: #tRüst.

tRüstHier:= ((CurrentTime) value) - tEIN.

tRüstNeu:= tRüstAlt + tRüstHier.

cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"

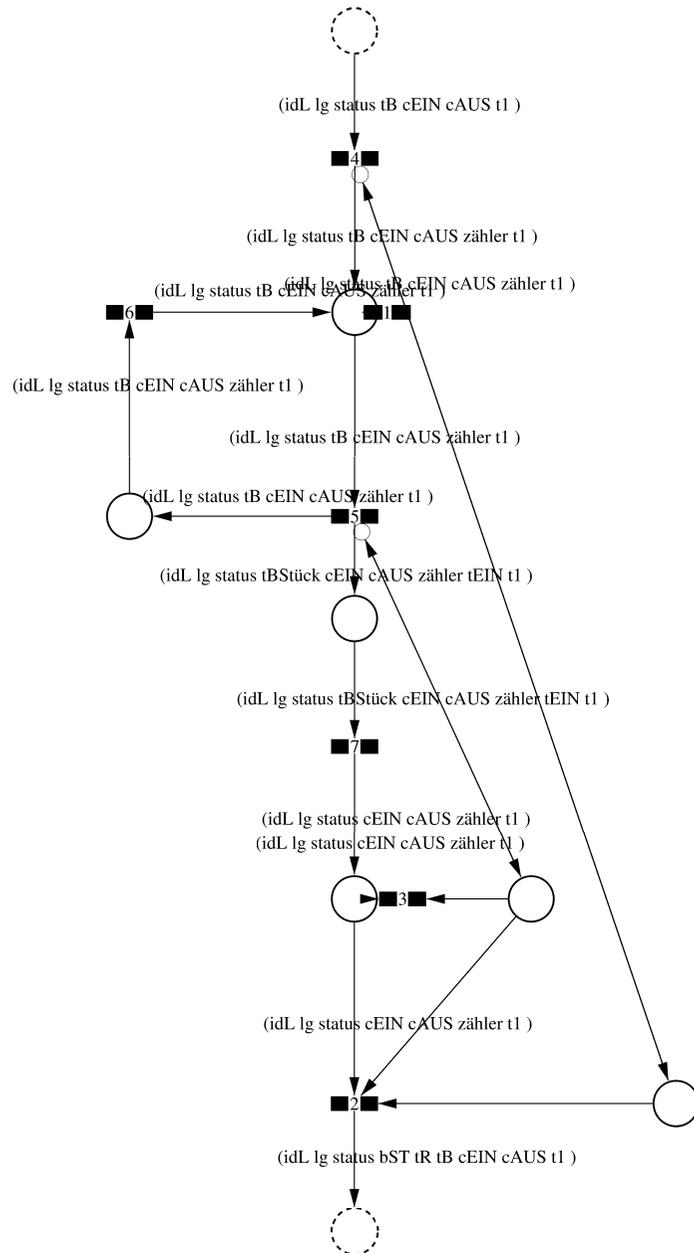
t2:= (CurrentTime) value.

rüstenAlt:= (self at: #rüsten) value.

rüstenNeu:= (t2-tEIN) +rüstenAlt.

(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
if True:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
if False:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"
zähler:= 1

5: Transition

====> Transition Variables <====

#zähler
#tBNetto
#tBVar
#tBStück
#tBVarProzent
#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"
tBNetto:= tB at: 1.
tBVarProzent:= tB at: 2.
(tBVarProzent > 0) if True:
 [tBVar:= ((tBVarProzent/100) *tBNetto).
 tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]
if False:
 [tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"
status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"
zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt
#tBearbHier
#tBearbNeu
#t2
#bearbeitenAlt
#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"
tBearbAlt:= cAUS at: #tBearb.
tBearbHier:= ((CurrentTime) value) - tEIN.
tBearbNeu:= tBearbAlt + tBearbHier.
cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
bearbeitenAlt:= (self at: #bearbeiten) value.
bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.
(self at: #bearbeiten) value: arbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.

Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"

anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

station:= (self at: #station) value.

Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.

(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu) printString.].

(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu) printString.].

(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu) printString.].

(station =#manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu) printString.].

(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu) printString.].

(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu) printString.].

(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu) printString.].

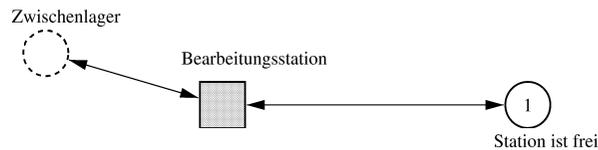
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu) printString.].

(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu) printString.].

(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu) printString.].

(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu) printString.].

Module: Prüfplätze



====> Development Comment <====

Modul: Kategorie Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Kategorie Bearbeitungsstation

Aufgabe: Dieses Modul "Kategorie Bearbeitungsstation" beinhaltet alle Bearbeitungsstationen EINES Typs (z.B. alle CNC-Drehmaschinen). Der Typ der Bearbeitungsstationen wird über die Modulvariable #maschine definiert. Die Anzahl der Bearbeitungsstationen in diesem Modul wird über die eingefügte Anzahl der Module "Bearbeitungsstation" eine Ebene tiefer definiert.

Anzahl der Eingangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Anzahl der Ausgangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Modulvariable Vorbesetzung (Initial) Beschreibung

#anzahl 0 Anzahl der Bearbeitungsstationen in diesem Modul.
#bearbeiten 0 Summiert die Bearbeitungszeiten der Bearbeitungsstationen während der Simulation auf.
#rüsten 0 Summiert die Rüstzeiten der Bearbeitungsstationen während der Simulation auf.
#station #eintragen Definition des Typs der Bearbeitungsstationen.
#warten 0 Summiert die Wartezeiten der Bearbeitungsstationen während der Simulation auf.

====> Net resp. Module Variables <====

#anzahl
#bearbeiten
#rüsten
#station

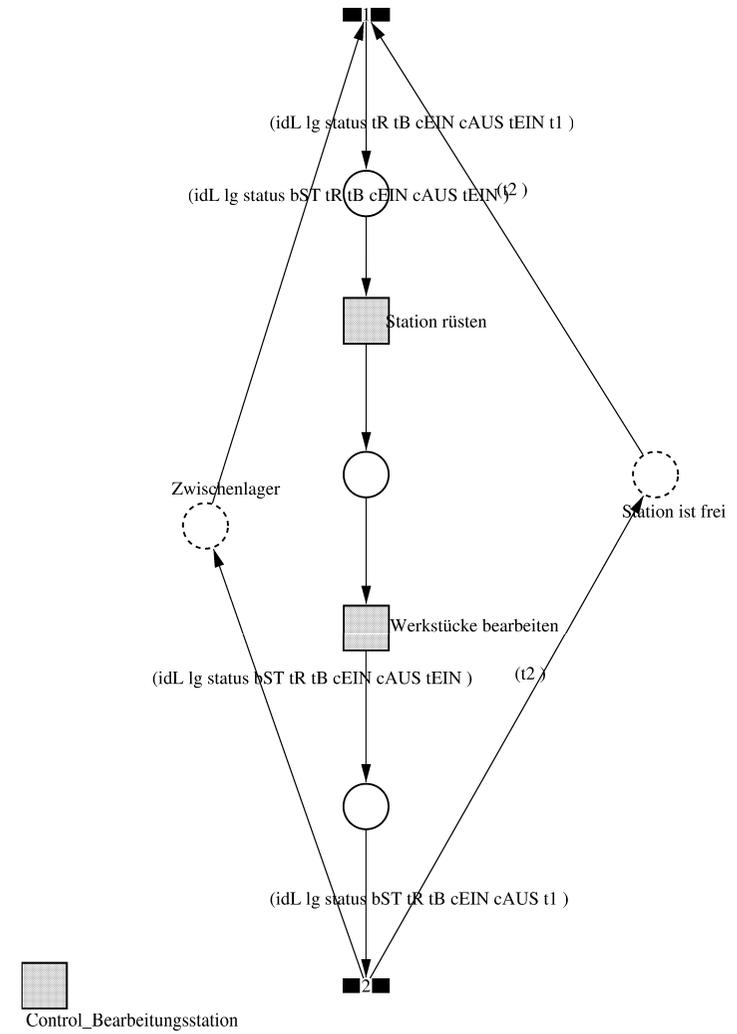
#warten

1: Place 'Station ist frei'

====> Initial Tokens <====

(CurrentTime)

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

```
#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel
```

====> Action Code <====

```
t2:= CurrentTime.
tEIN:= (CurrentTime) value.
```

```
"Status Los umsetzen"
status:= #wartet.
```

```
"Status der Bearbeitungsstation in Excel schreiben"
```

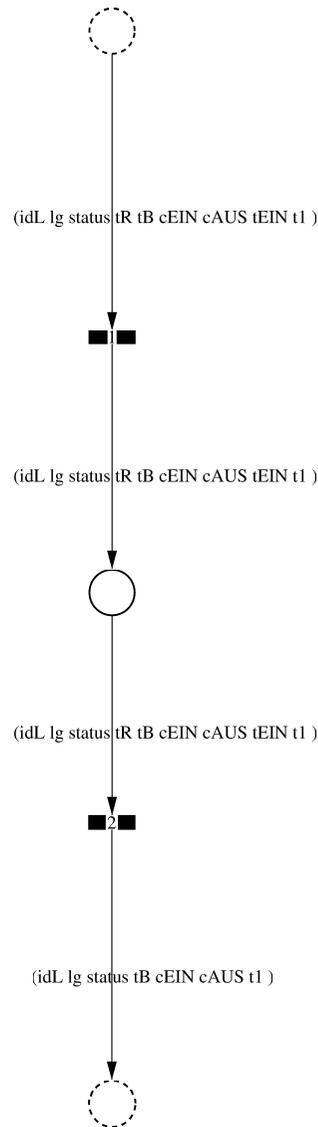
```
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:=(((rüsten*10) rounded)//10)
printString,',',(((rüsten*10)rounded)\10)*10)printString.
bearbeitenExcel:=(((bearbeiten*10) rounded)//10)
printString,',',(((bearbeiten*10)rounded)\10)*10)printString.
wartenExcel:=(((warten*10) rounded)//10)
printString,',',(((warten*10)rounded)\10)*10)printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.
```

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.].
(station = #manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].
(station = #manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].
(station = #gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].
(station = #montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].
(station = #prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].
(station = #extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.
    DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.
    DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"
```

```
betriebszeit:= CurrentTime value.
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)
printString,',',(((betriebszeit*10)rounded)\10)*10)printString.
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"
tRNetto:= tR removeFirst.
tRVarProzent:= tR removeFirst.
(tRVarProzent > 0) ifTrue:
[tRVar:= ((tRVarProzent/100) *tRNetto).
tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]
ifFalse:
[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

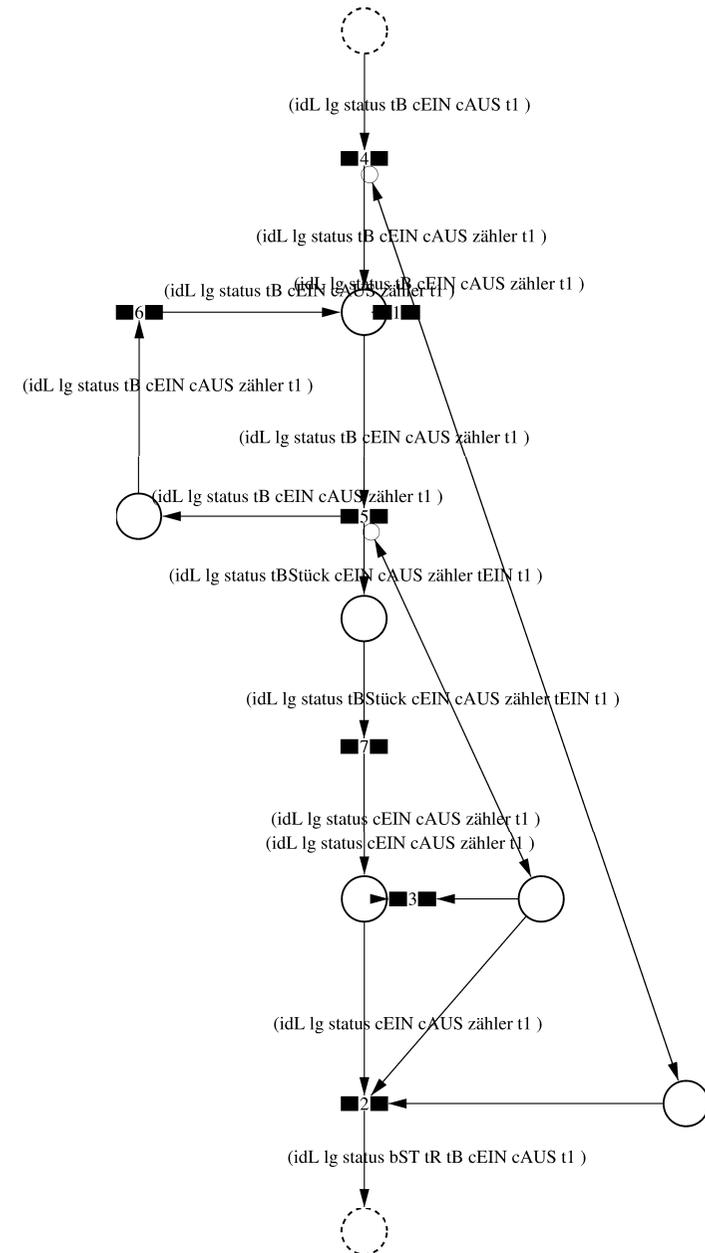
tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"
tRüstAlt:= cAUS at: #tRüst.
tRüstHier:= ((CurrentTime) value) - tEIN.
tRüstNeu:= tRüstAlt + tRüstHier.
cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
rüstenAlt:= (self at: #rüsten) value.
rüstenNeu:= (t2-tEIN) +rüstenAlt.
(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
ifTrue:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
ifFalse:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"

zähler:= 1

5: Transition

====> Transition Variables <====

#zähler

#tBNetto

#tBVar

#tBStück

#tBVarProzent

#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"

tBNetto:= tB at: 1.

tBVarProzent:= tB at: 2.

(tBVarProzent > 0) ifTrue:

[tBVar:= ((tBVarProzent/100) *tBNetto).

tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]

iffalse:

[tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"

status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"

zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt

#tBearbHier

#tBearbNeu

#t2

#bearbeitenAlt

#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"

tBearbAlt:= cAUS at: #tBearb.

tBearbHier:= ((CurrentTime) value) - tEIN.

tBearbNeu:= tBearbAlt + tBearbHier.

cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"

t2:= (CurrentTime) value.

bearbeitenAlt:= (self at: #bearbeiten) value.

bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.

(self at: #bearbeiten) value: bearbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.
Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

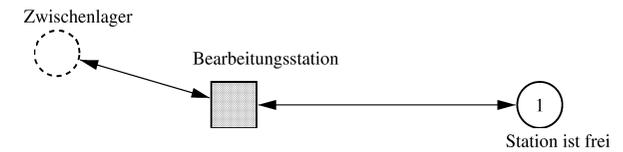
#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"
anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

```
station:= (self at: #station) value.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu)  
printString.].  
(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu)  
printString.].  
(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu)  
printString.].  
(station =#manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu)  
printString.].  
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu)  
printString.].  
(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu)  
printString.].  
(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu)  
printString.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu)  
printString.].  
(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu)  
printString.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu)  
printString.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu)  
printString.].
```

Module: Externe Bearbeitung



====> Development Comment <====

Modul: Kategorie Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Kategorie Bearbeitungsstation

Aufgabe: Dieses Modul "Kategorie Bearbeitungsstation" beinhaltet alle Bearbeitungsstationen EINES Typs (z.B. alle CNC-Drehmaschinen). Der Typ der Bearbeitungsstationen wird über die Modulvariable #maschine definiert. Die Anzahl der Bearbeitungsstationen in diesem Modul wird über die eingefügte Anzahl der Module "Bearbeitungsstation" eine Ebene tiefer definiert.

Anzahl der Eingangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Anzahl der Ausgangsstellen: 1
Bezeichner Attribute Beschreibung

Zwischenlager Beschrieben im tiefer liegenden Baustein "Bearbeitungsstation".

Modulvariable Vorbesetzung (Initial) Beschreibung

#anzahl 0 Anzahl der Bearbeitungsstationen in diesem Modul.
#bearbeiten 0 Summiert die Bearbeitungszeiten der Bearbeitungsstationen während der Simulation auf.
#rüsten 0 Summiert die Rüstzeiten der Bearbeitungsstationen während der Simulation auf.
#station #eintragen Definition des Typs der Bearbeitungsstationen.
#warten 0 Summiert die Wartezeiten der Bearbeitungsstationen während der Simulation auf.

====> Net resp. Module Variables <====

#anzahl
#bearbeiten
#rüsten
#station

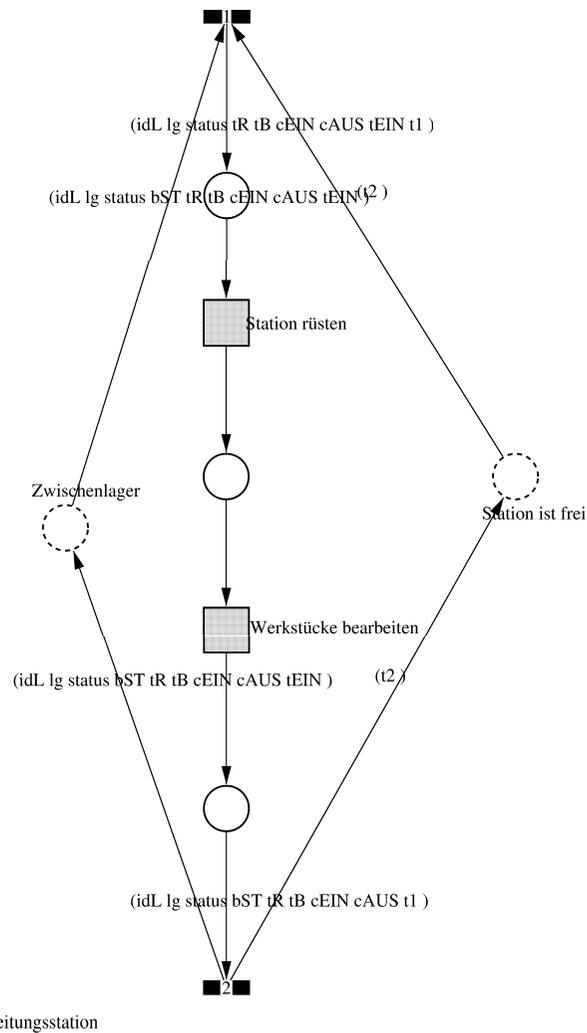
#warten

1: Place 'Station ist frei'

====> Initial Tokens <====

(CurrentTime)

Module: Bearbeitungsstation



====> Development Comment <====

Modul: Bearbeitungsstation
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Bearbeitungsstation

Aufgabe: Dieses Modul "Bearbeitungsstation" bildet eine Bearbeitungsstation ab. Das Modul ist universal aufgebaut und wird für jede Kategorie von Bearbeitungsstationen verwendet. Welcher Art die Bearbeitungsstation ist, wird über die Modulvariable #maschine im Modul "Kategorie Bearbeitungsstation" eine Ebene höher definiert.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	
Zwischenlager tEIN	Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.	
Station ist frei t2	Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.	

Anzahl der Ausgangsstellen: 2

Bezeichner Attribute Beschreibung

Bezeichner	Attribute	Beschreibung
Zwischenlager idL	Eindeutige Nummer des Loses, in Excel festgelegt.	
Zwischenlager lg	Anzahl der Werkstücke in dem Los, in Excel festgelegt.	
Zwischenlager status	Zeigt den aktuellen Status des Loses in der Simulation an.	
Zwischenlager bST	Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.	
Zwischenlager tR	Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.	
Zwischenlager tB	Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.	
Zwischenlager cEIN	Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.	
Zwischenlager cAUS	Collection, welche die Auswerte-Daten des Loses speichert.	

Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
Station ist frei t2 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tEIN
#tWarteAlt
#tWarteHier
#tWarteNeu
#t1
#wartenAlt
#wartenNeu
#bearbSt

====> Condition Code <====

bST= ((self at: #station) value).

====> Action Code <====

"Warten Los in cAUS schreiben"
tWarteAlt:= cAUS at: #tWarte.
tWarteHier:= ((CurrentTime) value) - tEIN.
tWarteNeu:= tWarteAlt + tWarteHier.
cAUS at: #tWarte put: tWarteNeu.

"Bearbeitungsstation in cAUS eintragen"
tEIN:= (CurrentTime) value.
bearbSt:= cAUS at: #bearbSt.
bearbSt addFirst: tEIN.
bearbSt addFirst: bST.
cAUS at: #bearbSt put: bearbSt.

"Status Los umsetzen"
status:= #rüstet.

"#warten Bearbeitungsstation aktualisieren"
t1:= (CurrentTime) value.
wartenAlt:= (self at: #warten) value.
wartenNeu:= (t1-t2) + wartenAlt.
(self at: #warten) value: wartenNeu.

2: Transition

====> Transition Variables <====

#bearbeiten
#bearbeitenExcel
#betriebszeit
#betriebszeitExcel
#rüsten
#rüstenExcel
#station
#tEIN
#warten
#wartenExcel

====> Action Code <====

t2:= CurrentTime.
tEIN:= (CurrentTime) value.

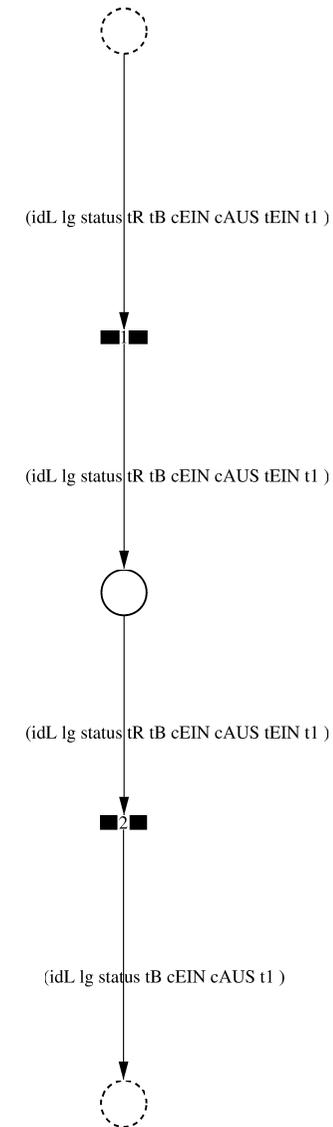
"Status Los umsetzen"
status:= #wartet.

"Status der Bearbeitungsstation in Excel schreiben"
station:= (self at: #station) value.
rüsten:= (self at: #rüsten) value.
bearbeiten:= (self at: #bearbeiten) value.
warten:= (self at: #warten) value.
rüstenExcel:= (((rüsten*10) rounded)//10)
printString, ', ', (((rüsten*10) rounded)\10)*10) printString.
bearbeitenExcel:= (((bearbeiten*10) rounded)//10)
printString, ', ', (((bearbeiten*10) rounded)\10)*10) printString.
wartenExcel:= (((warten*10) rounded)//10)
printString, ', ', (((warten*10) rounded)\10)*10) printString.
(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z2S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z2S4' data: wartenExcel.].
(station = #cncFräs) ifTrue: [DDE putData: Kanal3 item: 'Z3S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z3S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z3S4' data: wartenExcel.].
(station = #cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z4S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z4S4' data: wartenExcel.].
(station = #manDreh) ifTrue: [DDE putData: Kanal3 item: 'Z5S2' data: rüstenExcel.
DDE putData: Kanal3 item: 'Z5S3' data: bearbeitenExcel.
DDE putData: Kanal3 item: 'Z5S4' data: wartenExcel.].
(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S2' data: rüstenExcel.

```
DDE putData: Kanal3 item: 'Z6S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z6S4' data: wartenExcel.  
(station =#manBohr) ifTrue: [DDE putData: Kanal3 item: 'Z7S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z7S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z7S4' data: wartenExcel.].  
(station =#manSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z8S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z8S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z8S4' data: wartenExcel.].  
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z9S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z9S4' data: wartenExcel.].  
(station =#montage) ifTrue: [DDE putData: Kanal3 item: 'Z10S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z10S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z10S4' data: wartenExcel.].  
(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z11S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z11S4' data: wartenExcel.].  
(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S2' data: rüstenExcel.  
DDE putData: Kanal3 item: 'Z12S3' data: bearbeitenExcel.  
DDE putData: Kanal3 item: 'Z12S4' data: wartenExcel.].
```

```
"Betriebszeit in Excel schreiben"  
betriebszeit:= CurrentTime value.  
betriebszeitExcel:=(((betriebszeit*10) rounded)//10)  
printString, ', ',(((betriebszeit*10)rounded)\10)*10)printString.  
DDE putData: Kanal3 item: 'Z1S7' data: betriebszeitExcel.
```

Module: Station rüsten



====> Development Comment <====

Modul: Station rüsten
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Station rüsten

Aufgabe: In diesem Modul "Station rüsten" wird zunächst aus der Ordered Collection tR sowohl die Netto-Rüstzeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Rüstzeit tR bestimmt. Anschließend muss das Werkstück diese Zeit tR warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#tRNetto
#tR
#tRVar
#tRVarProzent

====> Action Code <====

"Rüstzeit und Schwankung auslesen und ausrechnen"

tRNetto:= tR removeFirst.

tRVarProzent:= tR removeFirst.

(tRVarProzent > 0) ifTrue:

[tRVar:= ((tRVarProzent/100) *tRNetto).

tR:= (Triangular from: (tRNetto - tRVar) to: (tRNetto + tRVar) shape: tRNetto) next.]

ifFalse:

[tR:= tRNetto].

2: Transition

====> Transition Variables <====

#tRüstAlt
#tRüstHier
#tRüstNeu
#rüstenAlt
#rüstenNeu
#t2

====> Delay Code <====

tR

====> Action Code <====

"Rüstzeit in cAUS schreiben"

tRüstAlt:= cAUS at: #tRüst.

tRüstHier:= ((CurrentTime) value) - tEIN.

tRüstNeu:= tRüstAlt + tRüstHier.

cAUS at: #tRüst put: tRüstNeu.

"#rüsten Bearbeitungsstation aktualisieren"

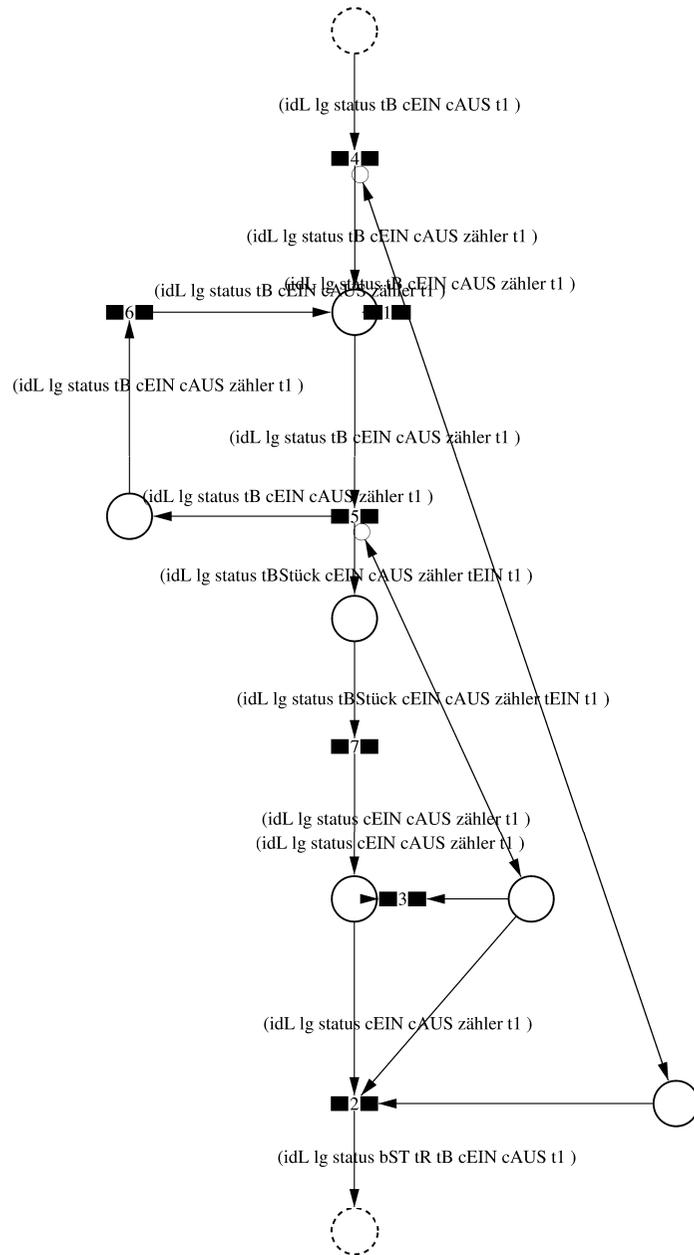
t2:= (CurrentTime) value.

rüstenAlt:= (self at: #rüsten) value.

rüstenNeu:= (t2-tEIN) +rüstenAlt.

(self at: #rüsten) value: rüstenNeu.

Module: Werkstücke bearbeiten



====> Development Comment <====

Modul: Werkstücke bearbeiten
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Werkstücke bearbeiten

Aufgabe: In diesem Modul "Werkstücke bearbeiten" wird zunächst aus der Ordered Collection tB sowohl die Netto-Bearbeitungszeit sowie eine evtl. eingegebene Schwankung herausgelesen und mit Hilfe einer Triangulären Verteilung die Bearbeitungszeit tB bestimmt. Anschließend muss das Werkstück diese Zeit tB warten.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

-
- idL Eindeutige Nummer des Loses, in Excel festgelegt.
- lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
- status Zeigt den aktuellen Status des Loses in der Simulation an.
- bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
- tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
- tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
- cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
- cAUS Collection, welche die Auswerte-Daten des Loses speichert.
- t1 Wird zur Auswertung der Bearbeitungs- und Rüstzeit benötigt.

1: Transition

====> Transition Variables <====

#zähler
#lg

====> Condition Code <====

zähler > lg

2: Transition

====> Transition Variables <====

#lg
#tB
#zähler
#bST

====> Condition Code <====

zähler = lg

====> Action Code <====

"Nächste Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
ifTrue:
 [bST:= cEIN removeFirst.
 tR:= cEIN removeFirst.
 tB:= cEIN removeFirst.]
ifFalse:
 [bST:= #ende.
 tR:= #ende.
 tB:=#ende.].

3: Transition

====> Transition Variables <====

#lg
#zähler

====> Condition Code <====

zähler < lg

4: Transition

====> Transition Variables <====

#zähler

====> Action Code <====

"Zähler festlegen"
zähler:= 1

5: Transition

====> Transition Variables <====

#zähler
#tBNetto
#tBVar
#tBStück
#tBVarProzent
#tEIN

====> Condition Code <====

zähler <= lg

====> Action Code <====

"Bearbeitungszeit und Schwankung auslesen und ausrechnen"
tBNetto:= tB at: 1.
tBVarProzent:= tB at: 2.
(tBVarProzent > 0) ifTrue:
 [tBVar:= ((tBVarProzent/100) *tBNetto).
 tBStück:= (Triangular from: (tBNetto - tBVar) to: (tBNetto + tBVar) shape: tBNetto) next.]
ifFalse:
 [tBStück:= tBNetto].

tEIN:= (CurrentTime) value.

"Status Los umsetzen"
status:= #bearbeitung.

6: Transition

====> Action Code <====

"Zähler aktualisieren"
zähler := zähler + 1

7: Transition

====> Transition Variables <====

#tBearbAlt
#tBearbHier
#tBearbNeu
#t2
#bearbeitenAlt
#bearbeitenNeu

====> Delay Code <====

tBStück

====> Action Code <====

"Bearbeitungszeit Los in cAUS eintragen"
tBearbAlt:= cAUS at: #tBearb.
tBearbHier:= ((CurrentTime) value) - tEIN.
tBearbNeu:= tBearbAlt + tBearbHier.
cAUS at: #tBearb put: tBearbNeu.

"#bearbeiten Bearbeitungsstation aktualisieren"
t2:= (CurrentTime) value.
bearbeitenAlt:= (self at: #bearbeiten) value.
bearbeitenNeu:= (t2-tEIN) +bearbeitenAlt.
(self at: #bearbeiten) value: arbeitenNeu.

Module: Control_Bearbeitungsstation

Anzahl eintragen



====> Development Comment <====

Modul: Control_Bearbeitungsstation
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Bearbeitungsstation

Aufgabe: Dieses Modul "Control_Bearbeitungsstation" liebt zunächst über die Modulvariable #anzahl des Moduls "Kategorie Bearbeitungsstation" (höher liegende Ebene) aus, wie viele Bearbeitungsstationen dieses Typs eingetragen wurden, und schreibt diese Anzahl in Excel.

Dieses Modul "Control_Bearbeitungsstation" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token und endet mit der Konsumierung des Token in der Transition.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place 'Anzahl eintragen'

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

#anzahlAlt
#anzahlNeu
#station

====> Action Code <====

"Anzahl Bearbeitungsstationen aus Modulvariablen auslesen, aktualisieren und in Excel schreiben"

anzahlAlt:= (self at:#anzahl) value.
anzahlNeu:= anzahlAlt + 1.
(self at: #anzahl) value: anzahlNeu.

station:= (self at: #station) value.

Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.

(station = #cncDreh) ifTrue: [DDE putData: Kanal3 item: 'Z2S5' data: (anzahlNeu) printString.].

(station = #cncFräs) ifTrue:[DDE putData: Kanal3 item: 'Z3S5' data: (anzahlNeu) printString.].

(station =#cncDrehFräs) ifTrue: [DDE putData: Kanal3 item: 'Z4S5' data: (anzahlNeu) printString.].

(station =#manDreh) ifTrue:[DDE putData: Kanal3 item: 'Z5S5' data: (anzahlNeu) printString.].

(station = #manFräs) ifTrue: [DDE putData: Kanal3 item: 'Z6S5' data: (anzahlNeu) printString.].

(station =#manBohr) ifTrue:[DDE putData: Kanal3 item: 'Z7S5' data: (anzahlNeu) printString.].

(station =#manSchleif) ifTrue:[DDE putData: Kanal3 item: 'Z8S5' data: (anzahlNeu) printString.].

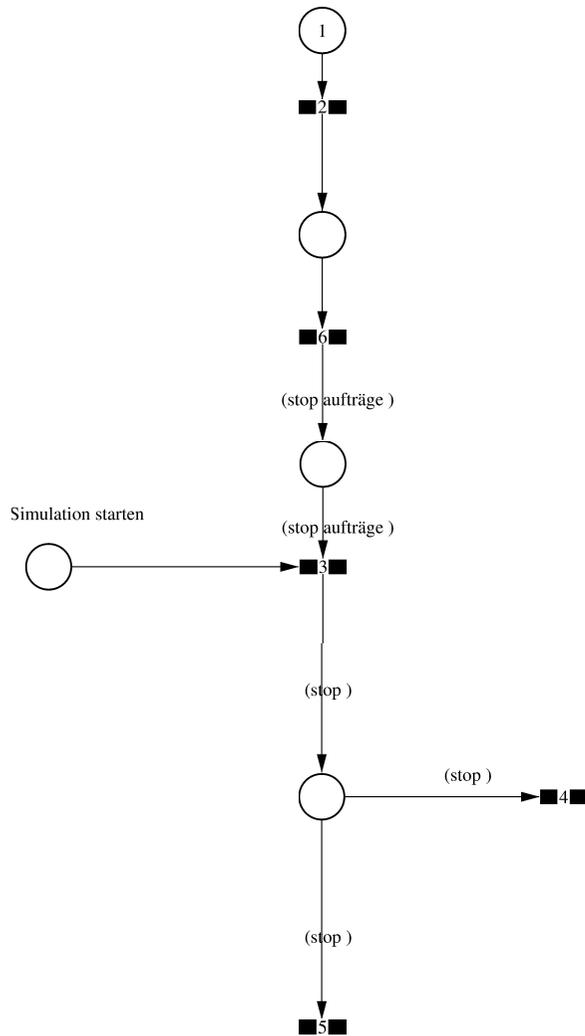
(station =#gleitSchleif) ifTrue: [DDE putData: Kanal3 item: 'Z9S5' data: (anzahlNeu) printString.].

(station =#montage) ifTrue:[DDE putData: Kanal3 item: 'Z10S5' data: (anzahlNeu) printString.].

(station =#prüf) ifTrue: [DDE putData: Kanal3 item: 'Z11S5' data: (anzahlNeu) printString.].

(station =#extern) ifTrue: [DDE putData: Kanal3 item: 'Z12S5' data: (anzahlNeu) printString.].

Module: Control_Start



====> Development Comment <====

Modul: Control_Start
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Control_Start

Aufgabe: Dieses Modul "Control_Start" prüft ob die Excel-Datei "Befüllung.xlsx" geöffnet ist, öffnet die Kanäle dorthin und überschreibt alte Ergebnisse in Excel mit "0".

Es wird die Anzahl der Aufträge sowie ein evtl. eingegebener vorzeitiger Stop der Simulation aus Excel eingelesen.

Anschließend werden so viele Token wie Aufträge in Excel eingegeben wurden in die Stelle "Start Aufträge" im Modul "Auftragspool" eingelegt.

Sobald die Aufträge im Modul "Auftragspool" eingelesen sind, wird von dort ein Token zurück in die Stelle "Simulation starten" dieses Moduls "Control_Start" gelegt.

Anschließend werden so viele Token in die Stelle "Start" gelegt wie Aufträge in Excel eingegeben wurden.

Im weiteren Verlauf ist dieses Modul nur noch relevant, falls ein vorzeitiger Stop der Simulation in Excel eingetragen wurde. In diesem Fall stoppt die Simulation nach der angegebenen Zeit.

Dieses Modul "Control_Start" ist ein Hilfsmodul und wird zu Beginn der Simulation ausgeführt. Es startet mittels eines Initial Token.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 0

1: Place

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

#abfrage
#testvariable

#aufträge
#excelAus
#container

====> Action Code <====

"Prüfen ob die Excel-Datei Befüllung.xlsx geöffnet ist"
abfrage:= DialogView confirm: 'Beim nun folgenden Start der Simulation kann es zu einer Wartezeit von bis zu einer Minute kommen, in der Sie keine Reaktion des Simulationswerkzeuges PACE sehen können. Bitte haben Sie Geduld, in dieser Zeit werden die Aufträge aus der Excel-Datei Befüllung.xlsx eingelesen.

Ist die Excel-Datei Befüllung.xlsx geöffnet?

,
initialAnswer: true.
(abfrage = false) ifTrue: [DialogView warn:'Bitte öffnen Sie zunächst die Excel-Datei und starten Sie den Lauf des PACE-Modells anschließend neu.'. self terminate.].
(abfrage = true) ifTrue: [
"Test ob wirklich offen"
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.
aufträge:= (DDE getData: Kanal3 item:'Z1S3') asNumber].

"Vorherige Ergebnisse aus Excel löschen, Blatt PACE_AUS_Masch; das MUSS hier geschehen, da ein Initial Token im Modul Control_Bearbeitungsstation direkt als nächstes läuft und die Anzahl der Bearbeitungsstationen einträgt!"
excelAus:= OrderedCollection new.
11 timesRepeat: [
container:= OrderedCollection new.
4 timesRepeat:[container addFirst: 0.].
excelAus addFirst: container.].
DDE writeNumbersToExcel: Kanal3 startAt: 'Z2S2' dataList: excelAus.
DDE putData: Kanal3 item: 'Z1S7' data: 0 printString.

3: Transition

====> Transition Variables <====

#startSimulation

====> Action Code <====

"Simulation starten"
startSimulation:= self placeNamed: 'Start'.
aufträge timesRepeat: [self addTokenTo: startSimulation].

4: Transition

====> Transition Variables <====

#stop

====> Condition Code <====

stop = 0

5: Transition

====> Condition Code <====

stop > 0

====> Delay Code <====

stop

====> Action Code <====

"Vorzeitigen Stop Simulation ausführen"
DialogView warn: 'Die Simulation wurde zu dem von Ihnen in Excel eingegeben Datum beendet.
Die Daten der Bearbeitungsstationen sind im Tabellenblatt Ergebnisse_Stationen in Excel ablesbar.

Möglicherweise wurden nicht alle Aufträge fertiggestellt - nicht fertiggestellte Aufträge werden
im Tabellenblatt Ergebnisse_Aufträge in Excel NICHT angezeigt.
Die Status dieser nicht fertiggestellten Aufträge können jedoch in PACE abgelesen werden.'

"Aktuelle Zeit als Ende Betriebszeit in Excel schreiben"
DDE putData: Kanal3 item: 'Z1S7' data: stop printString.
self terminate.

6: Transition

====> Transition Variables <====

#startAufträge
#abfrage
#aufträge

```
#excelAus  
#container  
#stop  
#startFrei  
#stoppFrei
```

====> Action Code <====

"Kanäle öffnen und Auftragsanzahl einlesen"

```
Kanal1:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS'.  
Kanal2:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_EIN'.  
Kanal3:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS_Masch'.  
aufträge:= (DDE getData: Kanal2 item:'Z1S3') asNumber.
```

"Vorherige Ergebnisse aus Excel löschen, Blatt PACE_AUS"

```
excelAus:= OrderedCollection new.  
100 timesRepeat: [  
  container:= OrderedCollection new.  
  27 timesRepeat:[container addFirst: 0].  
  excelAus addFirst: container.].  
DDE writeNumbersToExcel: Kanal1 startAt: 'Z4S1' dataList: excelAus.
```

"Prüfen ob Simulationsstart arbeitsfreier Tag"

```
startFrei:= (DDE getData: Kanal2 item:'Z1S7') asNumber.  
(startFrei = 99) ifTrue:[DialogView warn: '  
  Sie haben in der Excel-Tabelle für die Simulation keine Startzeit eingegeben.  
  Das Feld ist in Excel im Tabellenblatt Aufträge (oben) rot hinterlegt.  
  Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE  
  anschließend neu.'. self terminate.].  
(startFrei = 0) ifTrue:[DialogView warn: '  
  Sie haben in der Excel-Tabelle für die Simulation eine Startzeit gewählt,  
  die auf einen arbeitsfreien Tag fällt. Das Feld ist in Excel im Tabellenblatt Aufträge (oben)  
  rot hinterlegt.  
  Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE  
  anschließend neu.'. self terminate.].
```

"Evtl. vorzeitigen Stop Simulation lesen, prüfen ob arbeitsfreier Tag"

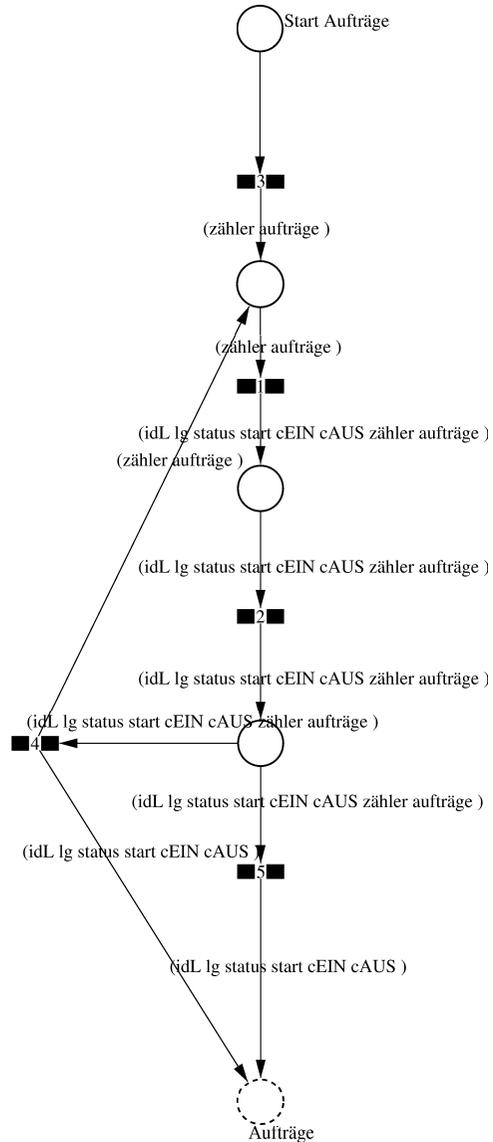
```
stop:= (DDE getData: Kanal2 item:'Z1S9') asNumber.  
stoppFrei:= (DDE getData: Kanal2 item:'Z1S10') asNumber.  
(stoppFrei = 0) ifTrue:[DialogView warn: '  
  Sie haben in der Excel-Tabelle für die Simulation eine Abbruchzeit gewählt, die auf einen  
  arbeitsfreien Tag fällt.  
  Das Feld ist in Excel rot hinterlegt.
```

Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE
anschließend neu.'. self terminate.].

"Marke in Auftragspool einlegen"

```
startAufträge:= self placeNamed:'Start Aufträge'.  
self addTokenTo: startAufträge.
```

Module: Auftragspool



====> Development Comment <====

Modul: Auftragspool
 Entwickler: Steffi Rudel
 Version: 1.0
 Datum: Januar 2016

====> Purpose Description <====

Modulname: Auftragspool

Aufgabe: In diesem Modul "Auftragspool" werden die Aufträge aus Excel als Lose eingelesen.

Anzahl der Eingangsstellen: 0

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

 Aufträge idL Eindeutige Nummer des Loses, in Excel festgelegt.
 Aufträge lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
 Aufträge status Zeigt den aktuellen Status des Loses in der Simulation an.
 Aufträge start Gibt die Simulationszeit an, zu der das Los startet.
 Aufträge cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
 Aufträge cAUS Collection, welche die Auswerte-Daten des Loses speichert.

1: Transition

====> Transition Variables <====

#z
 #s
 #tB
 #tR
 #tRNetto
 #tBNetto
 #hilfsvariable
 #bST
 #tBVarProzent
 #tRVarProzent

#status
#freiStart
#freiFertig

====> Action Code <====

"Aus Excel eingelesen werden:

idL Eindeutige Nummer des Loses
lg Anzahl der Werkstücke in dem Los
start Startzeit des Loses
bST Bearbeitungsstation (wird für jede Bearbeitungsstation gesondert eingelesen)
tRNetto Rüstzeit je Los (wird für jede Bearbeitungsstation gesondert eingelesen)
tRVarProzent Schwankung der Rüstzeit (wird für jede Bearbeitungsstation gesondert eingelesen)
tBNetto Bearbeitungszeit je Werkstück (wird für jede Bearbeitungsstation gesondert eingelesen)
tBVarProzent Schwankung der Bearbeitungszeit (wird für jede Bearbeitungsstation gesondert eingelesen)"

"Erzeugt werden:

status Zeigt den aktuellen Status des Loses in der Simulation an
cEIN Collection, welche tB und tR speichert
cAUS Collection, welche die Auswerte-Daten des Loses speichert.
tB Collection, welche tBNetto und tBVar speichert
tR Collection, welche tRNetto und tRVar speichert"

z:= zähler.
s:= 49.

"Aufträge aus Excel einlesen"

idL:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:1.
lg:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:2.
status:= #eingelesen.
start:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:3.
freiStart:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:54.
freiFertig:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:55.
cEIN:= OrderedCollection new.

(freiStart = 99) ifTrue: [DialogView warn: '

Sie haben in der Excel-Tabelle bei Los ', z printString, ' keine Startzeit eingetragen. Das Feld ist im Tabellenblatt Aufträge in Excel rot hinterlegt.

Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

(freiStart = 0) ifTrue: [DialogView warn: '

Sie haben in der Excel-Tabelle bei Los ', z printString, ' eine Startzeit gewählt, die auf einen arbeitsfreien Tag fällt.

Das Feld ist im Tabellenblatt Aufträge in Excel rot hinterlegt.

Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

(freiFertig = 0) ifTrue: [

DialogView warn: '

Sie haben in der Excel-Tabelle bei Los ', z printString, ' für die Soll-Fertigstellung ein Datum gewählt, das auf einen arbeitsfreien Tag fällt.

Das Feld ist im Tabellenblatt Aufträge in Excel rot hinterlegt.

Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

10 timesRepeat:

[
tB:= OrderedCollection new.
tR:= OrderedCollection new.
tBVarProzent:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at: s+4.
"Schwankung Bearbeitungszeit"
tBNetto:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at: s+3.
"Bearbeitungszeit"
tRVarProzent:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:s+2.
"Schwankung Rüstzeit"
tRNetto:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:s+1.
"Rüstzeit"
hilfsvariable:= ((DDE readNumbersFromExcel:Kanal2 item: 'Parameter')at: z)at:s.
"bST"

(hilfsvariable = 0)

ifTrue: [((tRNetto + tRVarProzent + tBNetto + tBVarProzent) ~= 0) ifTrue: [DialogView warn: '

Sie haben in der Excel-Tabelle bei Los ', z printString, ' zwar Rüst- bzw. Bearbeitungszeiten eingegeben, jedoch keine Bearbeitungsstation ausgewählt.

Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

s:=s-5.

]

ifFalse:[

((hilfsvariable = 8) or: (hilfsvariable = 11))

ifTrue: [((tBNetto+tBVarProzent) ~=0) ifTrue: [DialogView warn: '

Sie haben in der Excel-Tabelle bei Los ', z printString, ' die Bearbeitungsstation

Gleitschleifmaschine oder Extern ausgewählt.

Bitte geben Sie dort die gesamte Bearbeitungszeit des Loses unter der Rüstzeit ein, die Bearbeitungszeit/Stück bleibt frei.

Bitte korrigieren Sie diese Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

```
]
ifFalse: [((tBNetto = 0) and: (tBVarProzent~=0)) ifTrue: [DialogView warn: '
Sie haben in der Excel-Tabelle bei Los ', z printString, ' zwar eine Schwankung für die
Bearbeitungszeit eingegeben,
jedoch keine Bearbeitungszeit.
```

Bitte korrigieren Sie diese Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

```
]
((tRNetto = 0) and: (tRVarProzent~=0)) ifTrue: [DialogView warn: '
Sie haben in der Excel-Tabelle bei Los ', z printString, ' zwar eine Schwankung für die
Rüstzeit eingegeben, jedoch keine Rüstzeit.
```

Bitte korrigieren Sie diese Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

```
((tBNetto + tBVarProzent + tRNetto + tRVarProzent) =0)
ifTrue: [DialogView warn: '
Sie haben in der Excel-Tabelle bei Los ', z printString, ' zwar eine Bearbeitungsstation
ausgewählt, jedoch keine Rüst- bzw. Bearbeitungszeiten eingegeben.
```

Bitte prüfen und korrigieren Sie diese Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

```
ifFalse: [tR addFirst: tRVarProzent.
tR addFirst: tRNetto.
tB addFirst: tBVarProzent.
tB addFirst: tBNetto.
cEIN addFirst: tB.
cEIN addFirst: tR.
hilfsvariable = 0 ifTrue: [bST:= 0].
hilfsvariable = 1 ifTrue: [bST:= #cncDreh].
hilfsvariable = 2 ifTrue: [bST:= #cncFräs].
hilfsvariable = 3 ifTrue: [bST:= #cncDrehFräs].
hilfsvariable = 4 ifTrue: [bST:= #manDreh].
hilfsvariable = 5 ifTrue: [bST:= #manFräs].
hilfsvariable = 6 ifTrue: [bST:= #manBohr].
hilfsvariable = 7 ifTrue: [bST:= #manSchleif].
hilfsvariable = 8 ifTrue: [bST:= #gleitSchleif].
hilfsvariable = 9 ifTrue: [bST:= #montage].
hilfsvariable = 10 ifTrue: [bST:= #prüf].
hilfsvariable = 11 ifTrue: [bST:= #extern].
cEIN addFirst: bST.
```

```
]
s:=s-5.
].
].
cAUS:= Dictionary new.
```

2: Transition

====> Action Code <====

```
"Inhalte Container cAUS definieren"
cAUS at:#bearbSt put: OrderedCollection new.
cAUS at:#tEnde put:0.
cAUS at:#tBearb put: 0.
cAUS at:#tRüst put: 0.
cAUS at:#tWarte put: 0.
cAUS at:#tStart put: 0.
```

3: Transition

====> Transition Variables <====

```
#aufträge
#zähler
```

====> Action Code <====

```
"Auftragsanzahl aus Excel einlesen"
Kanal2:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_EIN'.
aufträge:= (DDE getData: Kanal2 item:'Z1S3') asNumber.
```

```
(aufträge=0) ifTrue: [DialogView warn: '
Sie haben in der Excel-Tabelle keine Aufträge eingetragen.
```

Bitte korrigieren Sie die Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.'. self terminate.].

```
"Zähler aktualisieren"
zähler:= 1.
```

4: Transition

====> Condition Code <====

```
zähler < aufträge
```

====> Action Code <====

"Zähler aktualisieren"
zähler:=zähler+1.

5: Transition

====> Transition Variables <====

#backControlSim

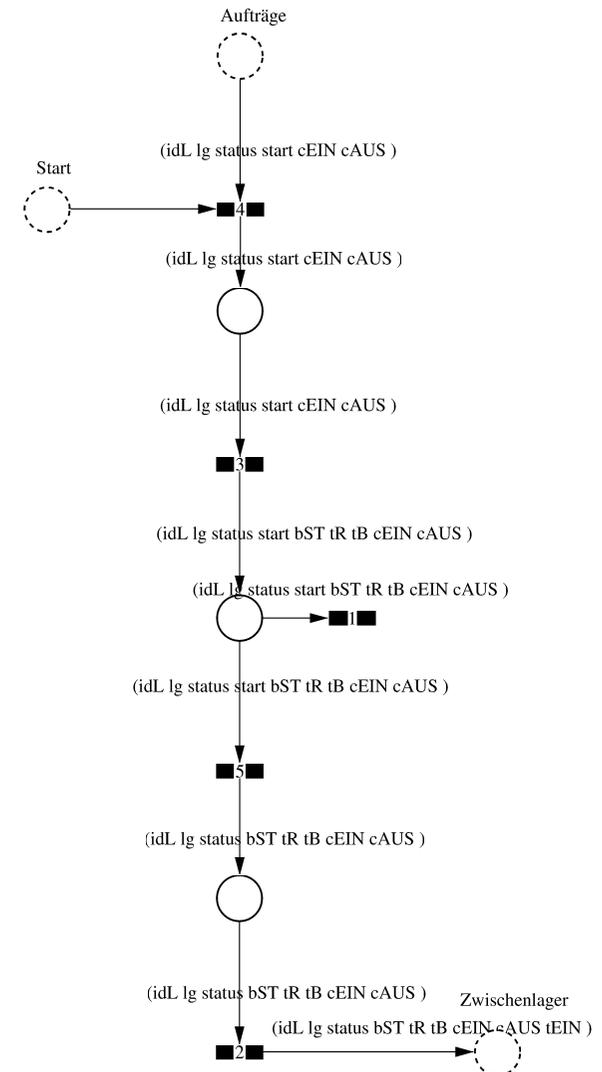
====> Condition Code <====

zähler =aufträge

====> Action Code <====

"Marke zurück ins Modul Control_Start legen"
backControlSim:= self placeNamed: 'Simulation starten'.
self addTokenTo: backControlSim.

Module: Arbeitsvorbereitung



====> Development Comment <====

Modul: Arbeitsvorbereitung
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Arbeitsvorbereitung

Aufgabe: In diesem Modul "Arbeitsvorbereitung" werden die Daten der Lose für die Simulation aufbereitet und die Lose in die Produktion gegeben.

Anzahl der Eingangsstellen: 2

Bezeichner Attribute Beschreibung

Start -- --

Aufträge idL Eindeutige Nummer des Loses, in Excel festgelegt.
Aufträge lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Aufträge status Zeigt den aktuellen Status des Loses in der Simulation an.
Aufträge start Gibt die Simulationszeit an, zu der das Los startet.
Aufträge cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Aufträge cAUS Collection, welche die Auswerte-Daten des Loses speichert.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.

1: Transition

====> Condition Code <====

start < 0

2: Transition

====> Transition Variables <====

#tEIN

====> Action Code <====

"tEIN eintragen"
tEIN:= (CurrentTime) value.

3: Transition

====> Transition Variables <====

#tB
#bST

====> Action Code <====

"1. Bearbeitungsstation aus cEIN rauslegen, sonst #ende eintragen"
cEIN notEmpty
if True:
[bST:= cEIN removeFirst.
tR:= cEIN removeFirst.
tB:= cEIN removeFirst.]
if False:
[bST:= #ende.
tR:= #ende.
tB:=#ende.].

4: Transition

====> Transition Variables <====

#container

====> Action Code <====

"Container ExcelMaschAus erzeugen und mit einer 0 füllen (löscht später alte Daten aus

Excel)"
ExcelMaschAus:= OrderedCollection new.
container:= OrderedCollection new.
container addFirst: 0.
ExcelMaschAus addFirst: container.

5: Transition

====> Condition Code <====

start >=0

====> Delay Code <====

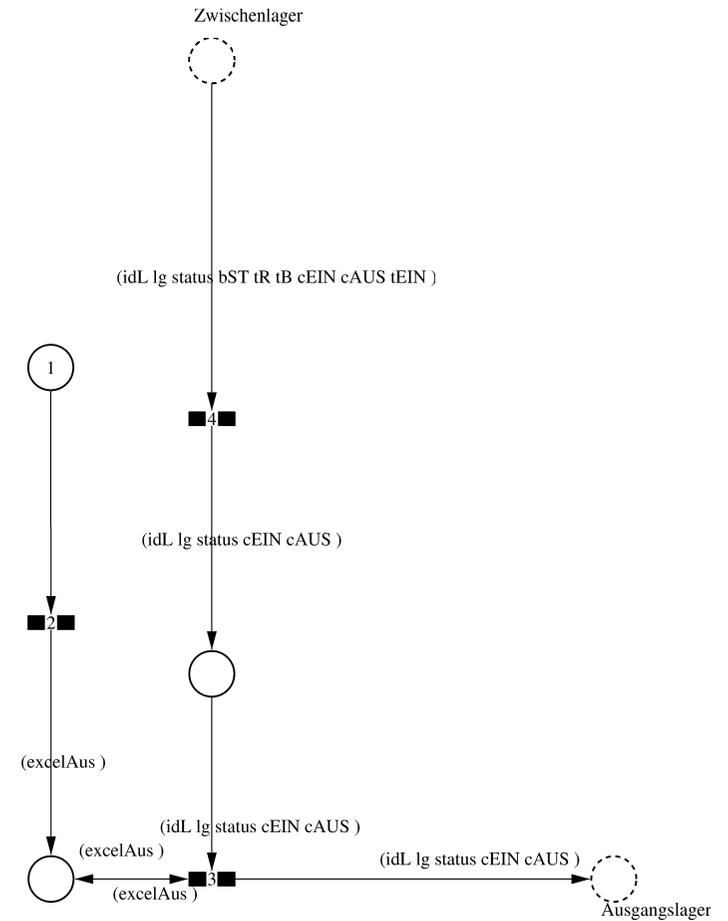
start

====> Action Code <====

"Status Los umsetzen"
status:= #wartet.

cAUS at: #tStart put: start.

Module: Arbeitsnachbereitung



====> Development Comment <====

Modul: Arbeitsnachbereitung
Entwickler: Steffi Rudel
Version: 1.0
Datum: Januar 2016

====> Purpose Description <====

Modulname: Arbeitsnachbereitung

Aufgabe: In dem Modul "Arbeitsnachbereitung" werden die Daten der simulierten Lose sowie der Bearbeitungsstationen zurück in Excel-Datei geschrieben.

Anzahl der Eingangsstellen: 1

Bezeichner Attribute Beschreibung

Zwischenlager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Zwischenlager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Zwischenlager status Zeigt den aktuellen Status des Loses in der Simulation an.
Zwischenlager bST Gibt an, auf welcher Bearbeitungsstation das Los bearbeitet wird.
Zwischenlager tR Gibt die Rüstzeit des Loses für die Bearbeitungsstation an.
Zwischenlager tB Gibt die Bearbeitungszeit je Werkstück für die Bearbeitungsstation an.
Zwischenlager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Zwischenlager cAUS Collection, welche die Auswerte-Daten des Loses speichert.
Zwischenlager tEIN Wird zur Auswertung der Wartezeit im Zwischenlager benötigt.

Anzahl der Ausgangsstellen: 1

Bezeichner Attribute Beschreibung

Ausgangslager idL Eindeutige Nummer des Loses, in Excel festgelegt.
Ausgangslager lg Anzahl der Werkstücke in dem Los, in Excel festgelegt.
Ausgangslager status Zeigt den aktuellen Status des Loses in der Simulation an.
Ausgangslager cEIN Collection, welche die Bearbeitungsstationen, Rüstzeiten und Bearbeitungszeiten des Loses speichert.
Ausgangslager cAUS Collection, welche die Auswerte-Daten des Loses speichert.

1: Place

====> Initial Tokens <====

()

2: Transition

====> Transition Variables <====

#excelAus
#container

====> Action Code <====

"Container erzeugen um Daten in Excel zu schreiben"
excelAus:= OrderedCollection new.
container:= OrderedCollection new.
container addFirst: 0.
excelAus addFirst: container.

3: Transition

====> Transition Variables <====

#tStart
#tWarte
#tRüst
#tBearb
#tEnde
#tDLZ.
#container
#a
#b
#bST
#tEIN
#bearbSt

====> Action Code <====

"Daten der Bearbeitungsstationen in Excel schreiben"
container:= OrderedCollection new.
bearbSt:= cAUS at: #bearbSt.
a:= bearbSt size.
b:= a/2.
b timesRepeat: [
bST:= bearbSt removeFirst.
tEIN:= bearbSt removeFirst.

```
container addFirst: tEIN.  
container addFirst: bST.  
].  
tDLZ:= cAUS at: #tDLZ.  
container addFirst: tDLZ.  
tEnde:= cAUS at: #tEnde.  
container addFirst: tEnde.  
tBearb:= cAUS at: #tBearb.  
container addFirst: tBearb.  
tRüst:= cAUS at: #tRüst.  
container addFirst: tRüst.  
tWarte:= cAUS at: #tWarte.  
container addFirst: tWarte.  
tStart:= cAUS at: #tStart.  
container addFirst: tStart.  
container addFirst: idL.  
excelAus add: container.  
Kanal1:= DDE open: 'EXCEL' topic: '[Befüllung.xlsx]', 'PACE_AUS'.  
DDE writeNumbersToExcel: Kanal1 startAt: 'Z3S1' dataList: excelAus.
```

4: Transition

====> Transition Variables <====

```
#bST  
#tDLZ  
#tStart  
#status
```

====> Condition Code <====

```
bST = #ende.
```

====> Action Code <====

```
"Durchlaufzeit in cAUS schreiben"  
cAUS at: #tEnde put:(CurrentTime) value.  
tStart:= cAUS at: #tStart.  
tDLZ:= ((CurrentTime) value) - tStart.  
cAUS at: #tDLZ put: tDLZ.
```

```
"Status Los umsetzen"  
status:= #fertig.
```

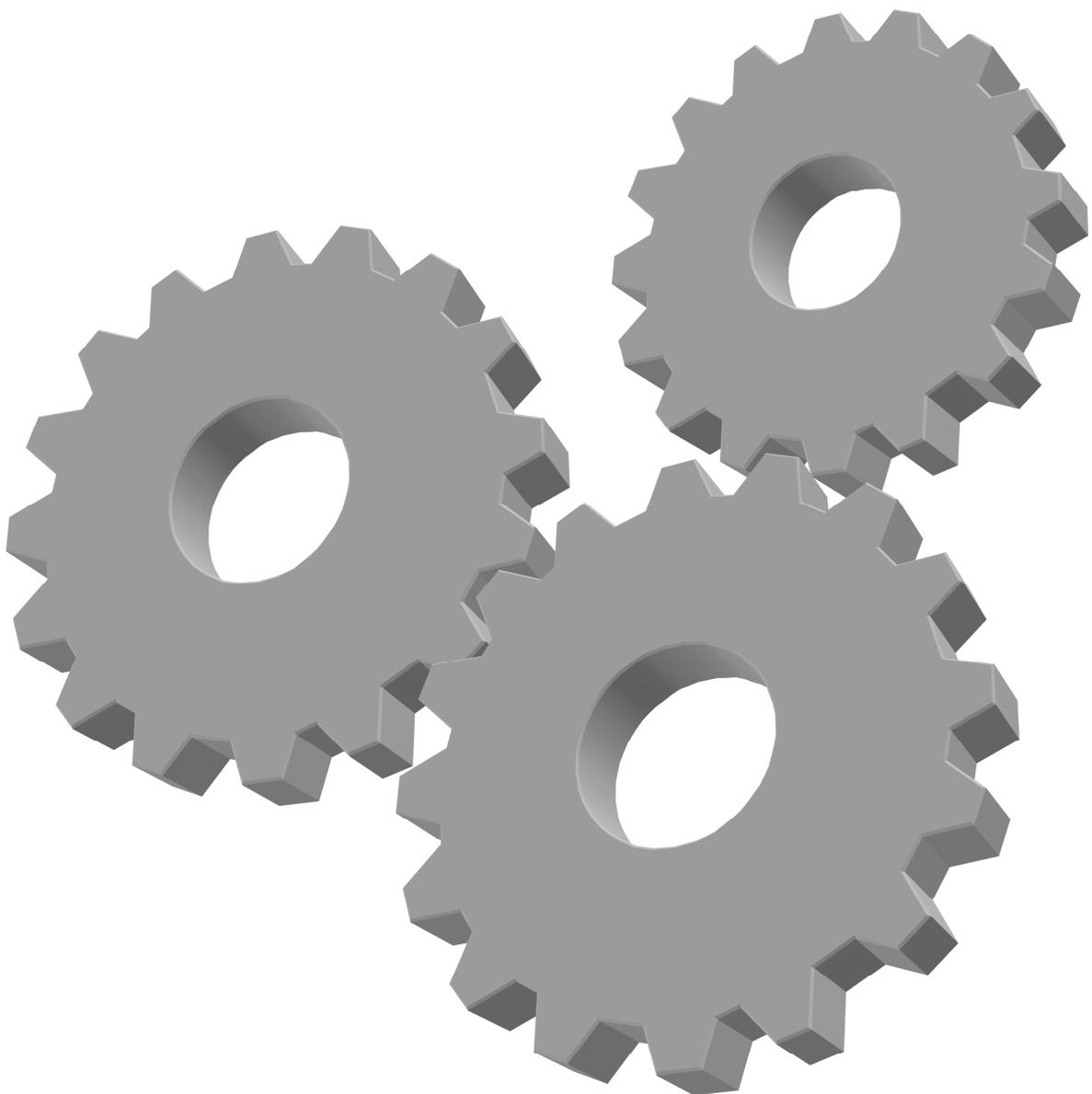
Anhang A8

Nutzerhandbuch

Nutzerhandbuch

Baukastensystem KMUSimMetall

Autor Steffi Rudel
Version 1.0
Datum 15.01.2016



Inhaltsverzeichnis

1	Warum und wofür?	1
2	Benötigte Programme	1
2.1	Das Simulationswerkzeug PACE.....	1
2.2	Die Microsoft-Office Anwendung Excel.....	1
3	Aufbau und Bestandteile des Baukastensystems	1
3.1	Aufbau.....	1
3.2	Bestandteile	2
3.2.1	Dateiordner Ergänzende Dateien	2
3.2.2	Dateiordner PACE-Bausteine.....	6
3.2.3	Text-Datei Version.txt.....	8
4	Vorbereitungen zur Nutzung des Baukastensystems	9
4.1	Wichtige Vorbemerkung zur Bedienung von PACE	9
4.2	PACE-Installation	9
4.3	Erstellen eines Simulationsmodelles in PACE	10
4.3.1	Erstellen eines neuen Netzes.....	10
4.3.2	Einfügen des Basisbausteins	11
4.3.3	Hinzuladen der Icons.....	13
4.4	Excel-Datei.....	14
5	Durchführen eines Simulationslaufes	14
5.1	Befüllen der Excel-Datei.....	14
5.1.1	Tabellenblatt Anmerkungen.....	14
5.1.2	Tabellenblatt Betriebszeiten	15
5.1.3	Tabellenblatt Aufträge	15
5.2	Simulationslauf starten	16
5.3	Simulationslauf beenden	17
5.4	Auswerten der Excel-Datei	18
5.4.1	Tabellenblatt Ergebnisse_Aufträge.....	18
5.4.2	Tabellenblatt Ergebnisse_Stationen	19
5.5	Speichern der Ergebnisse des Simulationslaufes.....	21
5.6	Simulationslauf im Hintergrund.....	21
6	Anpassen an Ihr Unternehmen.....	22
6.1	Jedes KMU ist anders	22
6.2	Löschen von Kategorien von Bearbeitungsstationen	22
6.3	Einfügen von Kategorien von Bearbeitungsstationen	22

Nutzerhandbuch Baukastensystem KMUSimMetall

6.4	Einfügen von Bearbeitungsstationen	24
6.5	Löschen von Bearbeitungsstationen.....	27
6.6	Speichern Ihres Simulationsmodelles.....	28
7	Schließen der Dateien.....	29
7.1	Schließen von PACE.....	29
7.2	Schließen von Excel.....	29

1 Warum und wofür?

Zunächst herzlichen Glückwunsch, dass Sie sich das **Baukastensystem KMUSimMetall** genauer ansehen. Alleine das zeigt, dass Sie daran interessiert sind, in Ihrem kleinen oder mittleren Unternehmen (KMU) der Branche „Herstellung von Metallerzeugnissen“ die Abläufe zu verbessern und wirtschaftlich zu arbeiten.

Das Baukastensystem unterstützt Sie mittels Prozess-Simulation bei der Planung Ihrer Produktion. Mit Prozess-Simulation können Sie, ohne die laufende Produktion unterbrechen zu müssen, beispielsweise folgende Fragen untersuchen:

- Welchen Liefertermin kann ich für den Auftrag zusagen?
- Wenn ich diesen zusätzlichen Auftrag annehme, wie verschieben sich die Fertigstellungstermine der anderen Aufträge?
- Wie lange ist die Durchlaufzeit/Rüstzeit/Bearbeitungszeit/Wartezeit des fertiggestellten Auftrages eigentlich gewesen?
- Welche Auslastung haben meine Maschinen?
- Welche meiner Maschinen stellt den „Flaschenhals“ dar?
- Welche Auswirkungen hätte es auf die Fertigstellungstermine der Aufträge, wenn wir die Arbeitszeit im Betrieb erhöhen oder reduzieren?
- Was würde es bringen, wenn ich eine zusätzliche Maschine des Typs XY anschaffe?
- Wie wirken sich (zusätzliche) freie Tage auf unsere Produktion aus?

2 Benötigte Programme

2.1 Das Simulationswerkzeug PACE

Um das Baukastensystem nutzen zu können, benötigen Sie das Simulationswerkzeug PACE. Dieses können Sie über die IBE Simulation Engineering GmbH unter www.ibepace.de beziehen.

PACE ist ein ausgereiftes, kompaktes, schnell zu installierendes Programm auf Petrinetz-Basis, welches keine besonderen Ansprüche an die Leistungsfähigkeit Ihres Computers stellt.

2.2 Die Microsoft-Office Anwendung Excel

Des Weiteren benötigen Sie die Microsoft-Office Anwendung Excel in der Version 2010 oder jünger.

3 Aufbau und Bestandteile des Baukastensystems

3.1 Aufbau

Das Baukastensystem beinhaltet die folgenden Elemente:

- Dateiordner **Ergänzende Dateien**
 - Excel-Datei **Befüllung.xlsx**
 - PACE-Datei **Icons.icn**

- dieses **Nutzerhandbuch.pdf**
- Dateiordner **PACE-Bausteine**
 - Dateiordner **Kategorien von Bearbeitungsstationen**
 - PACE-Baustein **CNC-Dreh-Fräszentren.sub**
 - PACE-Baustein **CNC-Drehmaschinen.sub**
 - PACE-Baustein **CNC-Fräsmaschinen.sub**
 - PACE-Baustein **Externe Bearbeitung.sub**
 - PACE-Baustein **Gleitschleifmaschinen.sub**
 - PACE-Baustein **Manuelle Bohrmaschinen.sub**
 - PACE-Baustein **Manuelle Drehmaschinen.sub**
 - PACE-Baustein **Manuelle Fräsmaschinen.sub**
 - PACE-Baustein **Manuelle Schleifmaschinen.sub**
 - PACE-Baustein **Montage.sub**
 - PACE-Baustein **Prüfplätze.sub**
 - PACE-Baustein **Basisbaustein.sub**
 - PACE-Baustein **Bearbeitungsstation.sub**
- Text-Datei **Version.txt**

Die Elemente (bis auf dieses Nutzerhandbuch) werden in den folgenden Kapiteln genauer beschrieben.

3.2 Bestandteile

3.2.1 Dateiordner Ergänzende Dateien

3.2.1.1 Excel-Datei Befüllung.xlsx

Die Excel-Datei besteht aus 6 Tabellenblättern.

Die Tabellenblätter sind in verschiedenen Farben gestaltet. Thematische zusammengehörende Tabellenblätter sind in der gleichen Farbe gestaltet.

Des Weiteren sind die Tabellenblätter teilweise in farbige und weiße Bereiche unterteilt. Die Tabellenblätter sind so geschützt, dass nur die weißen Bereiche ausgefüllt werden können.

Im Folgenden werden die Tabellenblätter im Einzelnen kurz beschrieben.

Nutzerhandbuch Baukastensystem KMUSimMetall

Im Tabellenblatt **Anmerkungen** sind Informationen und Einschränkungen aufgelistet, die Ihnen die Nutzung der Excel-Datei erleichtern sollen.

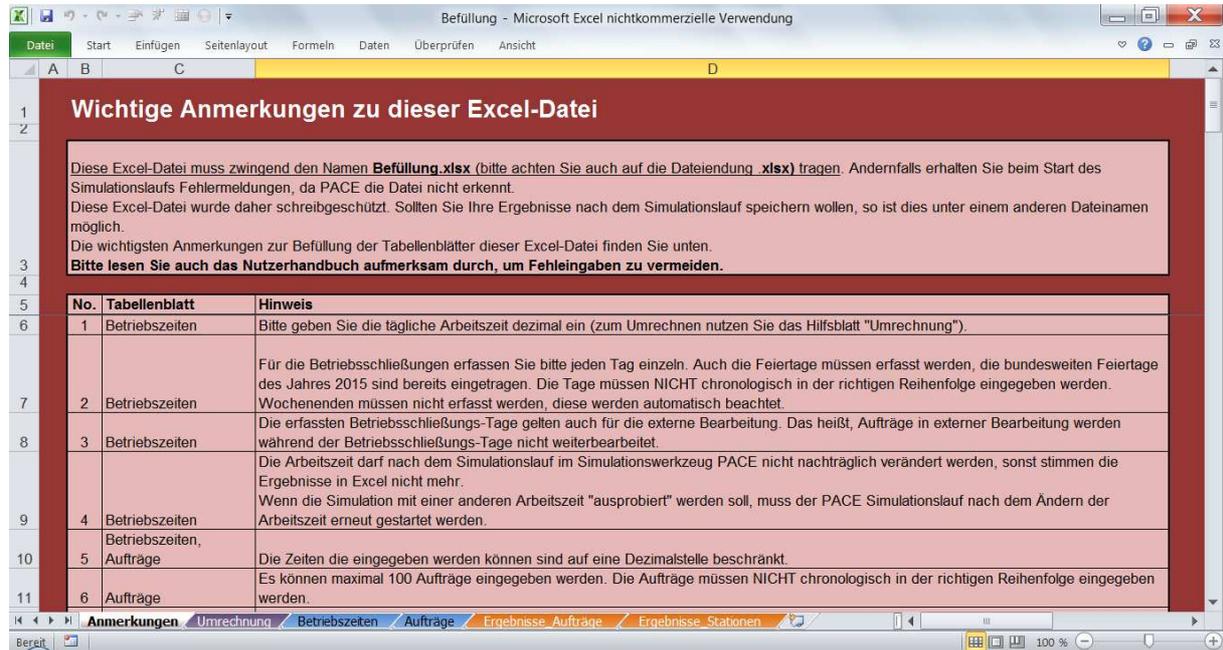


Abbildung 1: Tabellenblatt Anmerkungen

Im Tabellenblatt **Umrechnung** können Sie Bearbeitungszeiten in Dezimalzeiten sowie Tage in Minuten (dezimal) umrechnen.

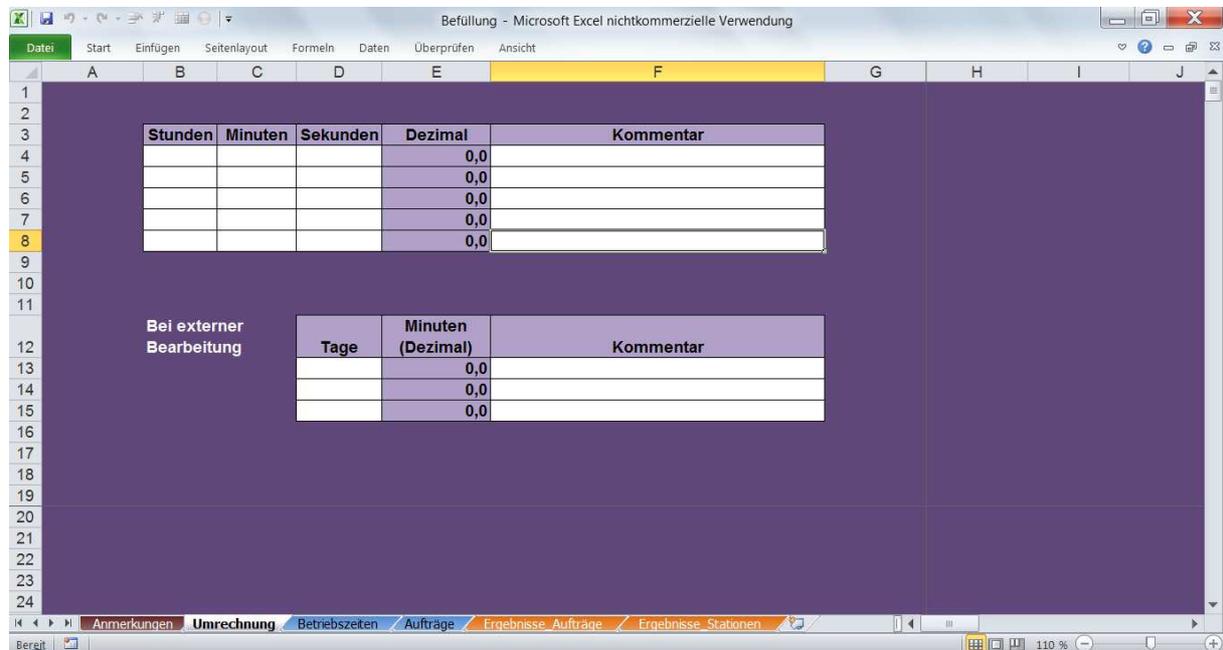


Abbildung 2: Tabellenblatt Umrechnung

Nutzerhandbuch Baukastensystem KMUSimMetall

Im Tabellenblatt **Betriebszeiten** geben Sie die Betriebszeiten sowie eventuelle Betriebs-schließungen ein.

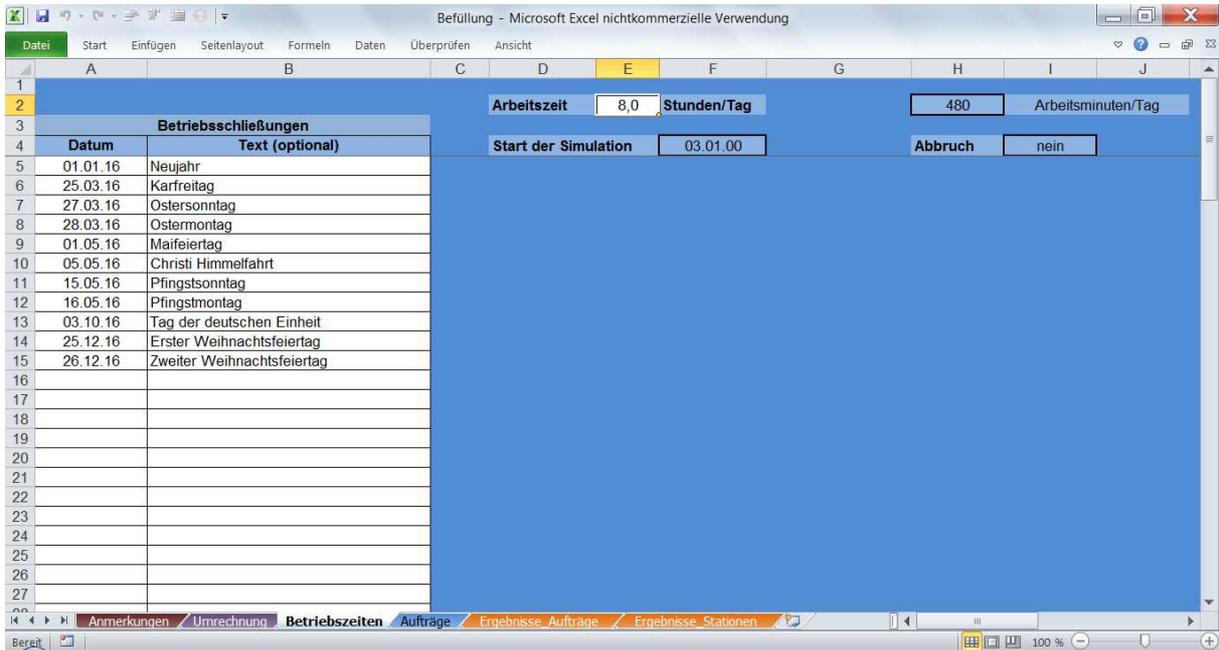


Abbildung 3: Tabellenblatt Betriebszeiten

Im Tabellenblatt **Aufträge** geben Sie die Daten der Aufträge ein. Des Weiteren geben Sie hier das Startdatum der Simulation ein. Optional können Sie zusätzlich ein Datum für den Abbruch des Simulationslaufes eingeben.

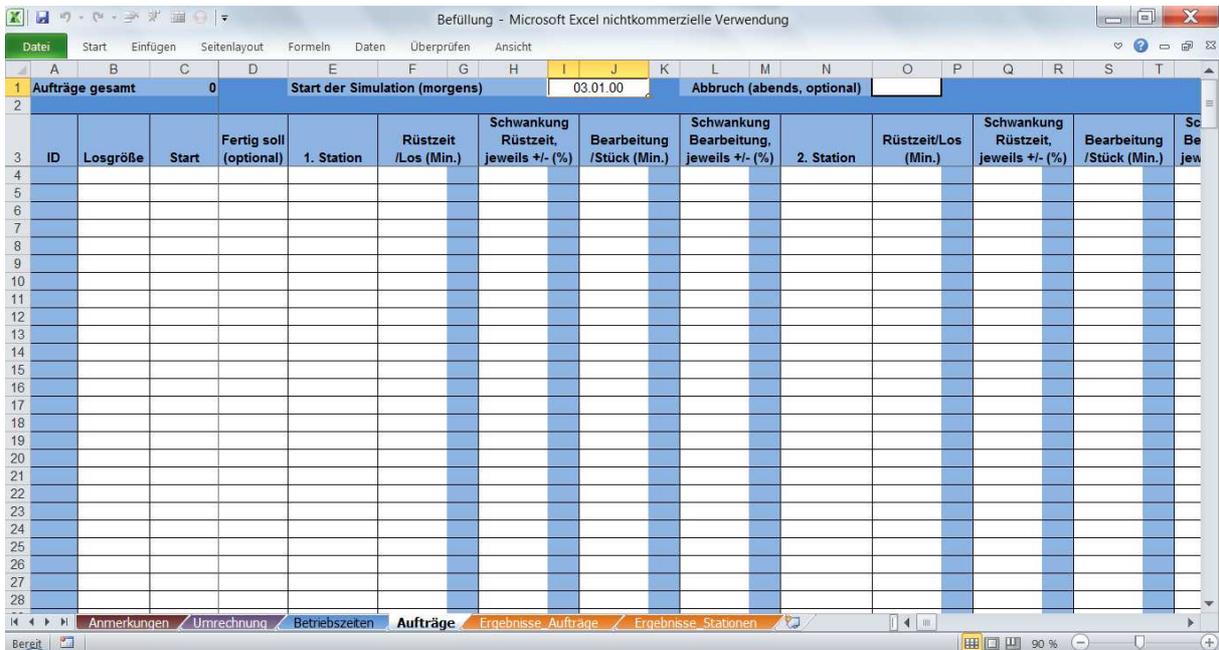


Abbildung 4: Tabellenblatt Aufträge

3.2.1.2 PACE-Datei Icons.icn

Die PACE-Datei **Icons.icn** beinhaltet die individuellen Icons des Baukastensystems. Die Datei muss manuell von PACE aus eingelesen werden.

Die folgende Abbildung zeigt den Inhalt der Datei.

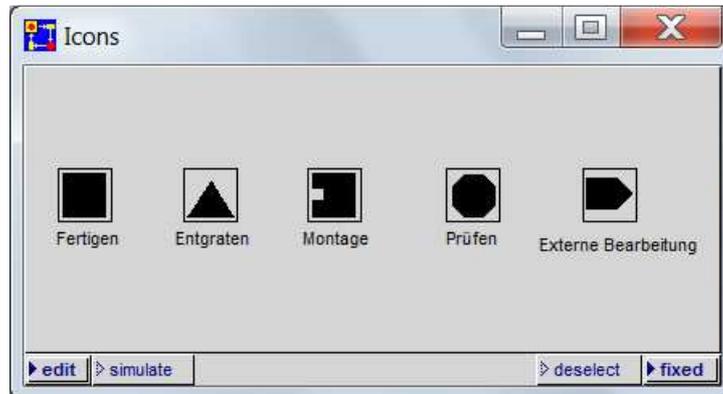


Abbildung 7: Individuelle Icons

3.2.2 Dateiordner PACE-Bausteine

3.2.2.1 Dateiordner Kategorien von Bearbeitungsstationen

In diesem Ordner ist jede mögliche Kategorie von Bearbeitungsstationen enthalten.

Die einzelnen Dateien dieses Ordners können zur Anpassung des Simulationsmodells an Ihr Unternehmen nach dem Einfügen des Basisbausteines in PACE hinzugeladen werden (Details dazu finden Sie in Kapitel 6 auf Seite 22).

3.2.2.2 PACE-Baustein Basisbaustein.sub

Der Basisbaustein beinhaltet, wie der Name schon sagt, die Basis Ihres Simulationsmodells. Der PACE-Baustein muss manuell von PACE aus eingelesen werden. Die Beschreibung dieses Vorgehens finden Sie in Kapitel 4.3.2 auf Seite 11.

Der Basisbaustein beinhaltet die Grundfunktionen des Simulationsmodells sowie jede mögliche Kategorie von Bearbeitungsstationen mit jeweils **einer** Station. Dieser Basisbaustein stellt damit das „Grundgerüst“ Ihres Simulationsmodells dar, das später an Ihr Unternehmen durch Entfernen einzelner Kategorien oder Hinzufügen von Bearbeitungsstationen angepasst werden kann (Details dazu finden Sie in Kapitel 6 auf Seite 22).

Die folgenden Abbildungen zeigen die Elemente des Basisbausteins.

Nutzerhandbuch Baukastensystem KMUSimMetall

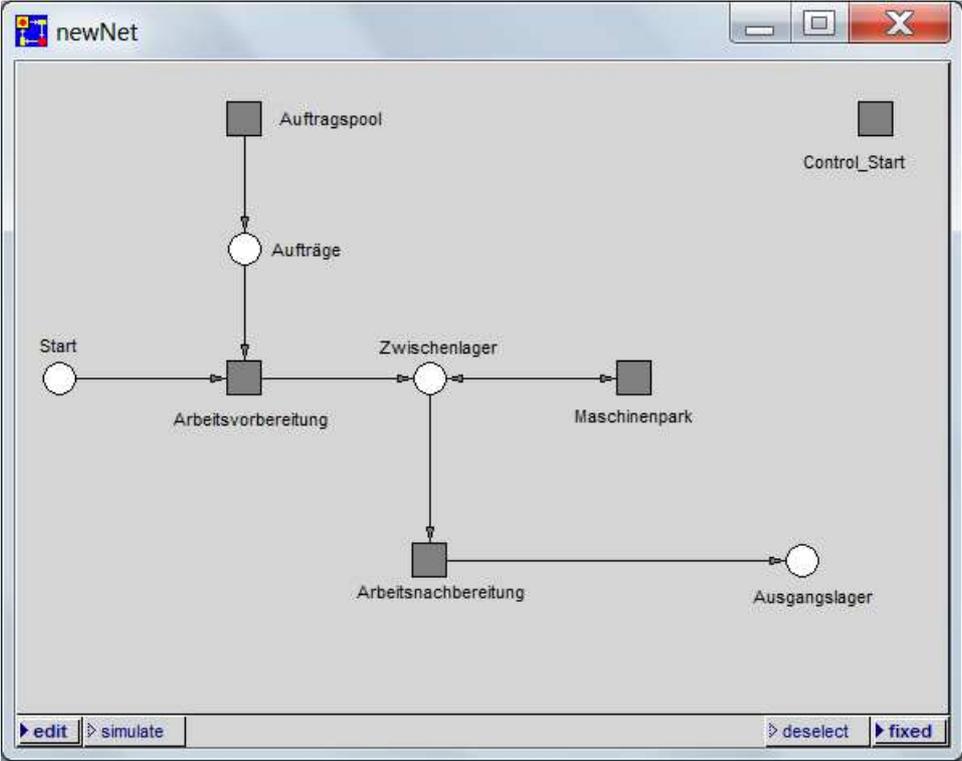


Abbildung 8: Elemente des Basisbausteins (1)

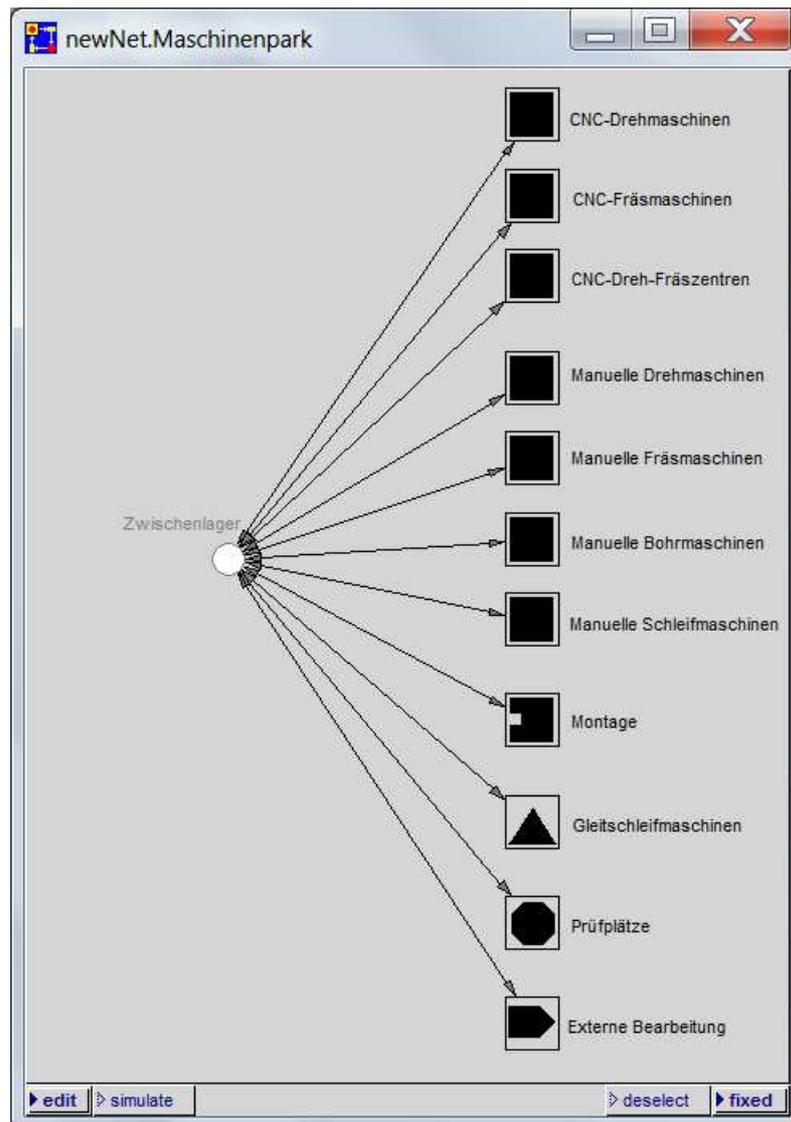


Abbildung 9: Elemente des Basisbausteins (2)

3.2.2.3 PACE-Baustein *Bearbeitungsstation.sub*

Die Datei **Bearbeitungsstation.sub** kann zur Anpassung des Simulationsmodells an Ihr Unternehmen nach dem Einfügen des Basisbausteines in PACE hinzugeladen werden (Details dazu finden Sie in Kapitel 6 auf Seite 22).

3.2.3 Text-Datei *Version.txt*

In dieser Datei sind einige allgemeine Informationen bezüglich des Baukastensystems hinterlegt:

- die Bezeichnung des Baukastensystems
- die Versionsnummer des Baukastensystems
- das Datum
- die Kontaktdaten der Ansprechpartnerin

4 Vorbereitungen zur Nutzung des Baukastensystems

4.1 Wichtige Vorbemerkung zur Bedienung von PACE

PACE besteht (anders als Sie das von gängigen Microsoft-Programmen gewohnt sind) aus einzelnen Programmfenstern. Diese können beliebig auf dem Bildschirm positioniert werden und unabhängig voneinander minimiert, maximiert oder geschlossen werden.

Um nicht jedes Mal wieder sämtliche Netzfenster maximieren zu müssen, wenn Sie im Wechsel mit der Excel-Datei arbeiten, empfiehlt es sich, ZUERST die PACE-Fenster und ANSCHLIEßEND die Excel-Datei zu öffnen. Nun können Sie die Excel-Datei beliebig minimieren und maximieren, während PACE im Hintergrund auf Ihrem Bildschirm geöffnet bleibt.

Bitte beachten Sie, dass Sie die PACE-Bedienleiste (beschriftet mit **PACE 2008 The IBE Simulator Developer**) sowie das Fenster **Net List** nicht schließen dürfen, es sei denn, Sie möchten das Simulationsmodell schließen und PACE beenden.

Die Mausbedienung in PACE ist anfangs etwas ungewohnt. Es wird neben der linken auch viel mit der rechten Maustaste gearbeitet. Häufig müssen Sie beim Aufruf von Befehlen die Maustaste drücken UND GEDRÜCKT HALTEN, bis Sie den gewünschten Menüpunkt ausgewählt haben. Durch das Loslassen der Maustaste wählen Sie dann den gewünschten Menüpunkt aus.

Zu Details lesen Sie bitte auch in der umfangreichen Dokumentation des Simulationswerkzeuges PACE nach, die zusammen mit PACE ausgeliefert wird.

4.2 PACE-Installation

Zur Installation von PACE öffnen Sie bitte die von IBE erhaltene Setup-Datei per Doppelklick. Ein InstallShield führt Sie durch die Installation. In der Regel dauert die Installation nur wenige Minuten.

Bitte achten Sie darauf die Installation in ein Windows-Verzeichnis vorzunehmen, auf das Sie (als Nutzer, NICHT NUR als Administrator!) vollen Lese- und Schreibzugriff haben (zu empfehlen ist ein Verzeichnis, in dem Sie auch Ihre Dokumente abspeichern). Andernfalls kann es bei der späteren Ausführung von PACE zu Fehlermeldungen des folgenden Formates kommen:



Abbildung 10: Fehlermeldung bei Problemen mit Lese- und Schreibzugriffen

Nach dem ersten Öffnen von PACE müssen Sie noch unter dem Menüpunkt **Help+Info** → **licence** → **insert licence** (bitte beachten Sie die etwas ungewohnte Mausbedienung in PACE, lesen Sie dazu auch in Kapitel 4.1 auf Seite 9 sowie in der umfangreichen Dokumentation des Simulationswerkzeuges PACE nach, die zusammen mit PACE ausgeliefert wird) Ihre gültige Lizenz eintragen (diese erhalten Sie ebenfalls über die IBE Simulation Engineering GmbH, siehe Kapitel 2.1 auf Seite 1) und mit Return abschließen. Die folgende Frage nach dem Speichern des Images bestätigen Sie mit **yes**.

4.3 Erstellen eines Simulationsmodelles in PACE

4.3.1 Erstellen eines neuen Netzes

Bitte starten Sie zunächst das Simulationswerkzeug PACE aus dem Windows-Startmenü.

Über den Menüpunkt **File** → **new net** erstellen Sie ein neues Netz. Den Namen für das neue Netz können Sie frei wählen, beispielsweise eignet sich Ihr Unternehmensname. Bestätigen Sie mit **OK**.



Abbildung 11: Neues Netz erstellen

Es öffnet sich ein Fenster, das „an Ihrem Mauszeiger klebt“. Sie können den Mauszeiger nun an eine beliebige Stelle Ihres Bildschirms bewegen und durch anschließendes Klicken mit der linken Maustaste das Fenster positionieren.

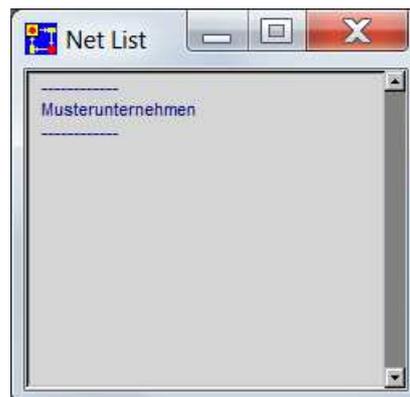


Abbildung 12: Net List

Das Fenster ist in der obersten Zeile als **Net List** bezeichnet. Klicken Sie in diesem Fenster **Net List** den von Ihnen gewählte Netznamen mit der linken Maustaste an. Während der Mauszeiger auf dem Netznamen steht, drücken Sie die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **edit** aus und lassen Sie die rechte Maustaste los.

Wieder öffnet sich ein Fenster, das „an Ihrem Mauszeiger klebt“. Sie können den Mauszeiger nun wieder an eine beliebige freie Stelle Ihres Bildschirms bewegen und durch anschließendes Klicken mit der linken Maustaste das Fenster positionieren.



Abbildung 13: Netzfenster

Dieses neue Netzfenster trägt in der obersten Zeile den von Ihnen vergebenen Netznamen (Sie können das Netzfenster durch Ziehen an den Ecken mit dem Mauszeiger vergrößern) und ist nun Ihre „Arbeitsfläche“, das heißt, in diesem Netzfenster erstellen Sie Ihr Simulationsmodell.

4.3.2 Einfügen des Basisbausteins

Positionieren Sie den Mauszeiger in Ihrem Netzfenster (vgl. Kapitel 4.3.1 auf Seite 10), drücken Sie die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **restore module** aus und lassen Sie die rechte Maustaste los. Es öffnet sich ein Windows-Explorer-Fenster, in dem Sie bitte aus dem Dateiordner **PACE-Bausteine** die Datei **Basisbaustein.sub** auswählen. Mit **Öffnen** bestätigen.

Es erscheint ein schwarzes Quadrat, das „an Ihrem Mauszeiger klebt“. Bewegen Sie den Mauszeiger in die Mitte Ihres Netzfensters und positionieren Sie den Basisbaustein durch Klicken mit der linken Maustaste. Es erscheint ein graues Quadrat, das als Basisbaustein beschriftet ist.

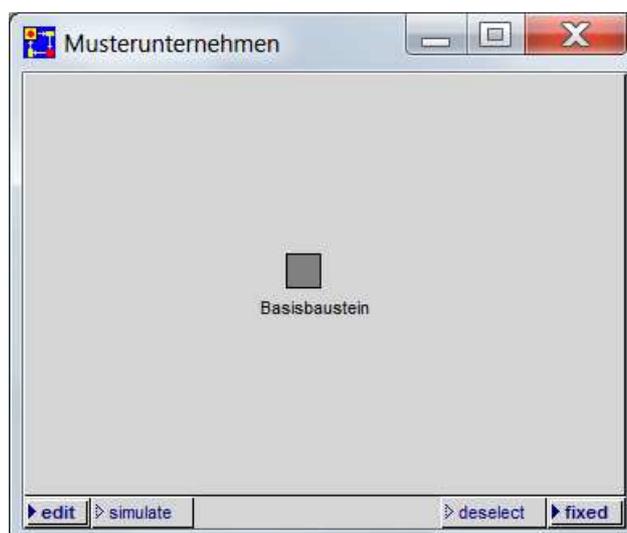


Abbildung 14: Eingefügter Basisbaustein

Nutzerhandbuch Baukastensystem KMUSimMetall

Klicken Sie dieses graue Quadrat mit der linken Maustaste mittig an (es erscheint ein kleines, schwarzes Quadrat in dem größeren, grauen Quadrat), drücken Sie die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **refine** aus und lassen Sie die rechte Maustaste los. Die anschließende Abfrage bestätigen Sie mit **yes**. PACE entpackt jetzt den Basisbaustein. Wenn alles geklappt hat, sollte Ihr Netzfenster nun so aussehen:

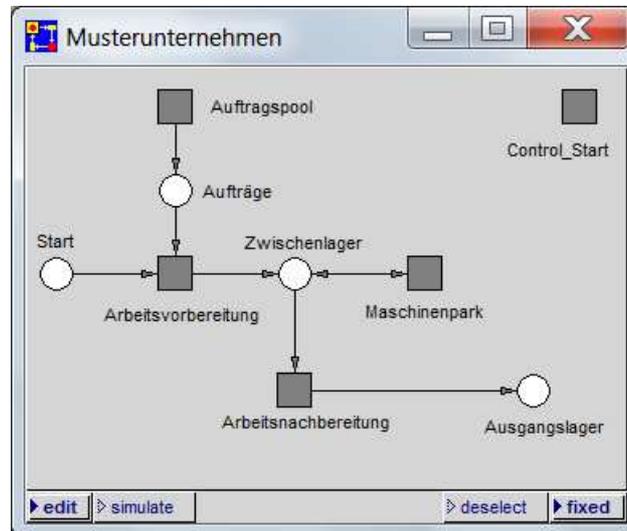


Abbildung 15: Entpackter Basisbaustein

Klicken Sie nun mit der linken Maustaste in das graue Quadrat, das mit Maschinenpark beschriftet ist (es erscheint ein kleines, schwarzes Quadrat in dem größeren, grauen Quadrat). Drücken Sie die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **subnet** aus und lassen Sie die rechte Maustaste los. Wieder öffnet sich ein Fenster, das „an Ihrem Mauszeiger klebt“. Bewegen Sie den Mauszeiger rechts neben Ihr Netzfenster und positionieren Sie das neue Fenster Maschinenpark. Anschließend ziehen Sie das Fenster Maschinenpark so groß, dass alle Bezeichnungen gut lesbar sind (klicken Sie dazu auf eine Ecke des Netzfensters Maschinenpark und ziehen Sie bei gedrückter Maustaste das Fenster größer). Ihr Bildschirm sollte nun so aussehen:

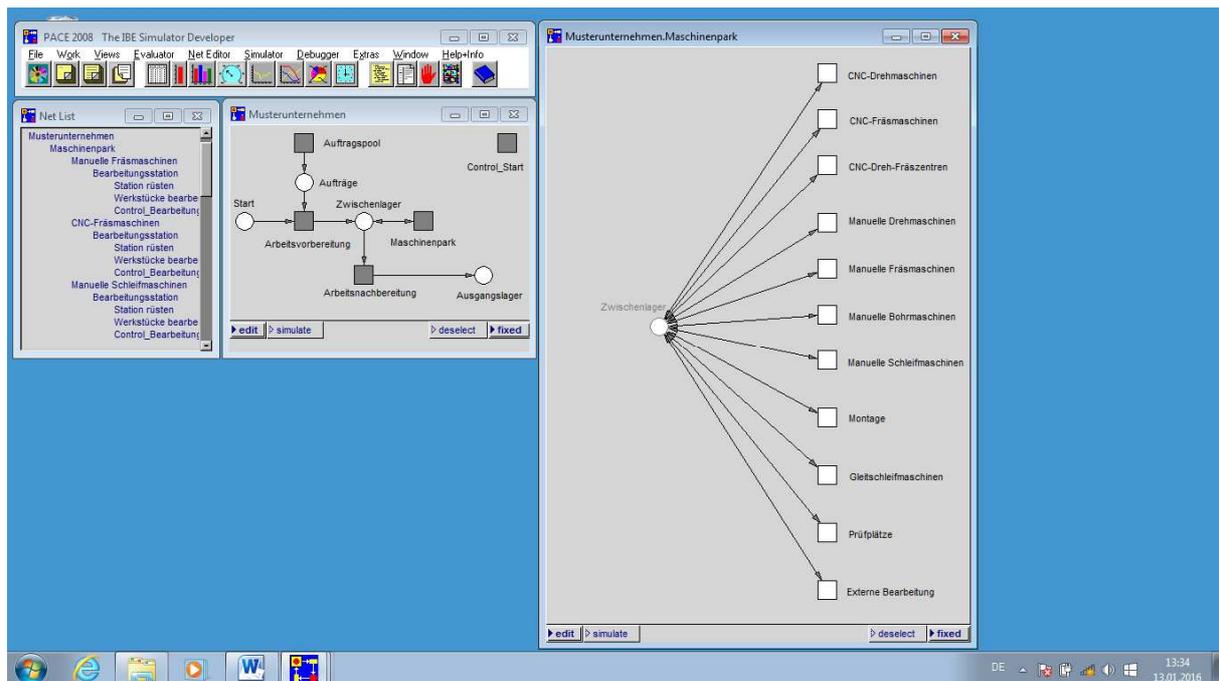


Abbildung 16: Bildschirmansicht (1)

4.3.3 Hinzuladen der Icons

Damit die Kategorien der Bearbeitungsstationen in Ihrem Netzfenster Maschinenpark noch mit den individuellen Icons dargestellt werden, müssen Sie abschließend die Datei **Icons.icn** hinzuladen. Dazu rufen Sie in der PACE-Bedienleiste (beschriftet mit **PACE 2008 The IBE Simulator Developer**) bitte den Menüpunkt **Extras** auf. Wählen Sie mit der linken Maustaste den Menüpunkt **icons** UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **load icon file** aus und lassen Sie die linke Maustaste los. Es öffnet sich ein Windows-Explorer-Fenster, in dem Sie bitte aus dem Dateiordner **Ergänzende Dateien** die Datei **Icons.icn** auswählen. Mit **Öffnen** bestätigen. Das Ergebnis sollte nun so aussehen:

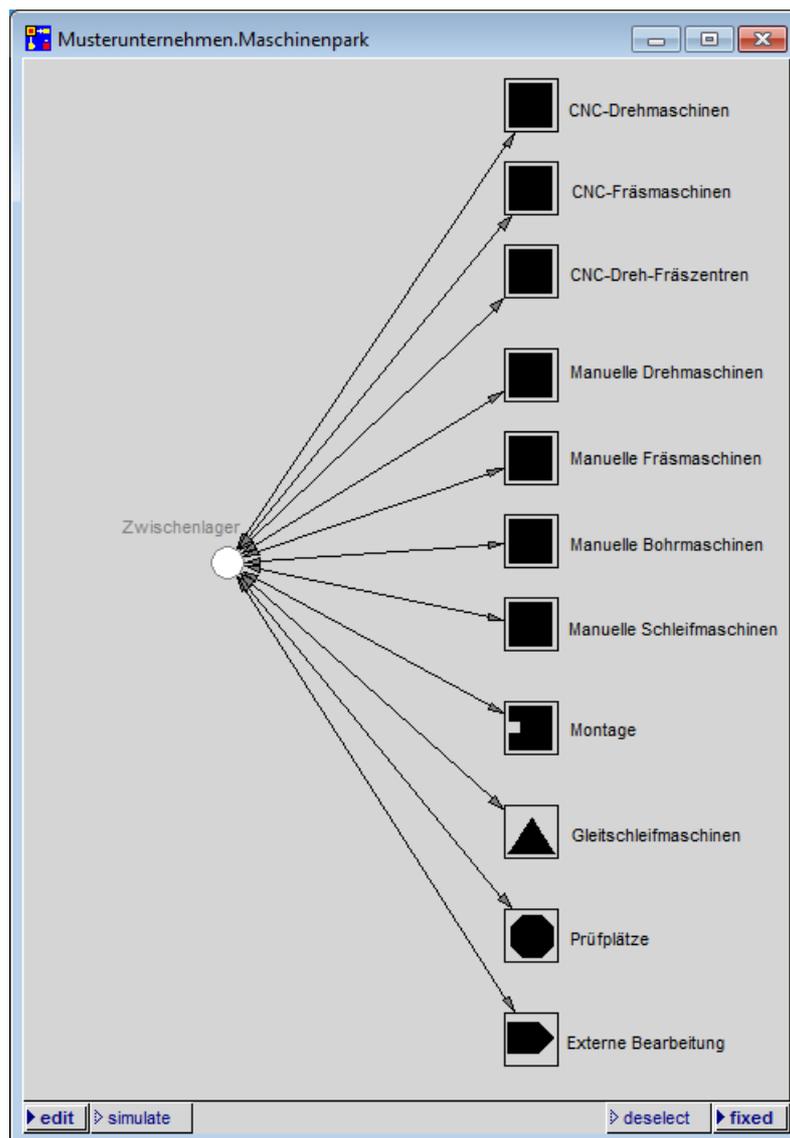


Abbildung 17: Maschinenpark mit individuellen Icons

Damit ist das Simulationsmodell vorerst fertig und kann im nächsten Schritt an Ihr Unternehmen angepasst (siehe Kapitel 6 ab Seite 22) oder direkt abgespeichert (siehe Kapitel 6.6 auf Seite 28) werden.

4.4 Excel-Datei

Zur Befüllung des Simulationsmodells und zur Auswertung des Simulationslaufes benötigen Sie noch die Excel-Datei **Befüllung.xlsx**. Bitte öffnen Sie diese per Doppelklick.

5 Durchführen eines Simulationslaufes

5.1 Befüllen der Excel-Datei

5.1.1 Tabellenblatt Anmerkungen

Bitte lesen Sie zunächst die Anmerkungen zu der Excel-Datei im Tabellenblatt **Anmerkungen** UND BEACHTEN SIE DIESE. Andernfalls können beim Simulationslauf immer wieder Fehler auftreten.

Nutzerhandbuch Baukastensystem KMUSimMetall

5.1.2 Tabellenblatt Betriebszeiten

Rufen Sie das Tabellenblatt **Betriebszeiten** auf und geben Sie die tägliche Arbeitszeit Ihres Betriebes sowie eventuelle Betriebsschließungen ein.

Beachten Sie dabei, dass die Betriebszeiten in Stunden/Tag (dezimal) eingegeben werden. Das Tabellenblatt **Umrechnung** hilft gegebenenfalls bei der Umrechnung in Dezimalzeiten.

Für die Betriebsschließungen erfassen Sie bitte jeden Tag einzeln. Auch die Feiertage müssen erfasst werden, die bundesweiten Feiertage des Jahres 2015 sind bereits eingetragen. Die Tage müssen NICHT chronologisch in der richtigen Reihenfolge eingegeben werden. Wochenenden müssen nicht erfasst werden, diese werden automatisch als arbeitsfrei beachtet.

5.1.3 Tabellenblatt Aufträge

Rufen Sie das Tabellenblatt **Aufträge** auf und geben Sie zunächst das gewünschte Startdatum des Simulationslaufs ein. Optional können Sie zusätzlich ein Datum für einen vorzeitigen Abbruch des Simulationslaufes eingeben.

Anschließend geben Sie bitte die spezifischen Daten der Aufträge ein (jeder Auftrag erhält eine eigene Zeile, es können maximal 100 Aufträge eingegeben werden). Bitte achten Sie darauf, in der obersten Zeile (Zeile 4) zu beginnen.

Im Folgenden sind die auszufüllenden Felder von links nach rechts beschrieben:

Losgröße

Bitte geben Sie eine Zahl zwischen 1 und 50000 ein (es können maximal 50000 Teile pro Auftrag eingegeben werden).

Start

Bitte geben Sie das Datum ein, zu dem der Auftrag in die Produktion gegeben werden soll. Bitte beachten Sie das Datumsformat TT.MM.JJ.

Aufträge, die vor dem eingegebenen Startdatum des Simulationslaufes starten sollen, werden beim Simulationslauf ignoriert. Das Startdatum betreffender Aufträge wird **gelb hinterlegt**.

Die **rote Hinterlegung** von einem eingegebenen Datum bedeutet, dass es auf einen arbeitsfreien Tag fällt. Bitte korrigieren Sie in diesem Fall Ihre Angabe auf einen Arbeitstag.

Im Folgenden können Sie Daten für bis zu 10 chronologisch aufeinander folgende Bearbeitungsstationen eingeben.

Fertig soll (optional)

Hier können Sie optional eingeben, wann der Auftrag fertig bearbeitet sein soll. Bitte beachten Sie das Datumsformat TT.MM.JJ.

Die **rote Hinterlegung** von einem eingegebenen Datum bedeutet, dass es auf einen arbeitsfreien Tag fällt. Bitte korrigieren Sie in diesem Fall Ihre Angabe auf einen Arbeitstag.

Station

Nutzerhandbuch Baukastensystem KMUSimMetall

Bitte wählen Sie aus dem Pulldown-Auswahlmenü eine Bearbeitungsstation aus.

Rüstzeit/Los (Min.)

Bitte geben Sie die Rüstzeit des Loses in Minuten (dezimal) ein. Die Rüstzeit kann auch 0 betragen.

Falls Sie die Bearbeitungsstation Gleitschleifmaschine oder externe Bearbeitung gewählt haben, geben Sie bitte die gesamte Bearbeitungszeit in diesem Feld Rüstzeit/Los (Min.) ein. Das Feld Bearbeitung/Stück (Min.) bleibt in diesem Fall frei.

Schwankung Rüstzeit, jeweils +/- (%)

Hier können Sie angeben, ob die von Ihnen eingegebene Rüstzeit schwanken soll. Sie können aus einem Pulldown-Auswahlmenü Werte zwischen 0% und 100% auswählen. Das Feld kann frei bleiben, das entspricht dann einem Wert von 0%.

Beispiel:

Falls Sie für die Rüstzeit eines Loses auf einer Station einen Wert von 2,0 Minuten angegeben haben und nun eine Schwankung der Rüstzeit von 50% auswählen (dies entspricht 1,0 Minuten), werden die Rüstzeiten dieses Loses auf dieser Station während des Simulationslaufes zufallsverteilt im Bereich zwischen 1,0 und 3,0 Minuten schwanken.

Bearbeitung/Stück (Min.)

Bitte geben Sie die Bearbeitungszeit des Loses in Minuten (dezimal) ein.

Falls Sie die Bearbeitungsstation Gleitschleifmaschine oder externe Bearbeitung gewählt haben, geben Sie bitte die gesamte Bearbeitungszeit unter Rüstzeit/Los (Min.) ein. Das Feld Bearbeitung/Stück (Min.) bleibt in diesem Fall frei.

Schwankung Bearbeitung, jeweils +/- (%)

Hier können Sie angeben, ob die von Ihnen eingegebene Bearbeitungszeit schwanken soll. Sie können aus einem Pulldown-Auswahlmenü Werte zwischen 0% und 100% auswählen. Das Feld kann frei bleiben, das entspricht dann einem Wert von 0%.

Beispiel:

Falls Sie für die Bearbeitungszeit eines Werkstückes auf einer Station einen Wert von 0,5 Minuten angegeben haben und nun eine Schwankung der Bearbeitungszeit von 100% auswählen (dies entspricht 0,5 Minuten), werden die Bearbeitungszeiten dieses Werkstückes auf dieser Station während des Simulationslaufes zufallsverteilt im Bereich zwischen 0,0 und 1,0 Minuten schwanken.

5.2 Simulationslauf starten

Um den Simulationslauf in PACE zu starten, minimieren Sie bitte das Excel-Fenster und aktivieren Sie in allen Netzfenstern von PACE die Einstellung **simulate** (jeweils links am unteren Rand des Netzfensters).

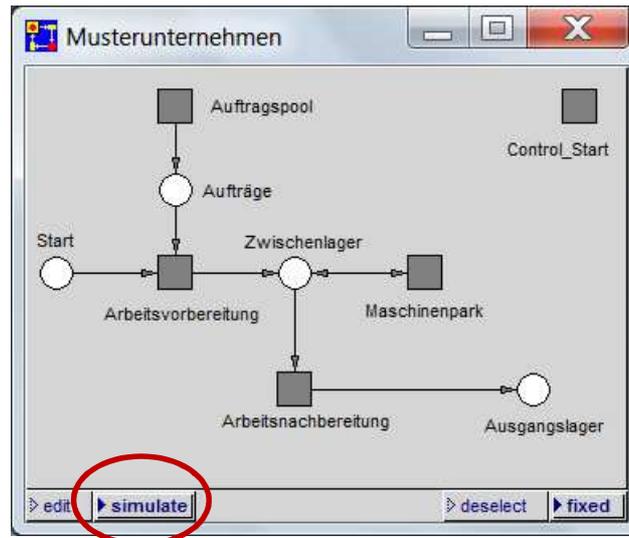


Abbildung 18: Einstellung simulate gewählt

Anschließend positionieren Sie den Mauszeiger in einem Netzfenster, drücken Sie die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **initialize + run** aus und lassen Sie die rechte Maustaste los.

Es erscheint ein Fenster mit dem Hinweis, dass beim folgenden Start des Simulationslaufes eine Wartezeit von bis zu einer Minute entstehen kann, in der Sie keine Reaktion von PACE erkennen können. Außerdem wird abgefragt, ob die Excel-Datei **Befüllung.xlsx** geöffnet ist. (Bitte achten Sie auch auf die richtige Dateiergung **.xlsx**! Sollten Sie die Datei z.B. als **Befüllung.xls** abgespeichert haben, erhalten Sie beim Start des Simulationslaufes Fehlermeldungen, da PACE die Datei nicht erkennt.)

Bitte prüfen Sie, ob die Excel-Datei **Befüllung.xlsx** geöffnet ist und beantworten Sie die Frage mit **yes**.

Falls alles klappt, sollte der Simulationslauf nun starten. Sie erkennen das daran, dass sich schwarze „Marken“ in den Netzfenstern hin- und herbewegen.

Unter Umständen erhalten Sie Hinweisfenster, dass in Excel etwas falsch eingetragen wurde oder ein Eintrag fehlt. Bitte bestätigen Sie in diesem Fall das Hinweisfenster sowie das folgende Hinweisfenster „End of simulation“ mit **OK**, korrigieren Sie Ihre Eingaben in Excel und starten Sie den Simulationslauf in PACE anschließend neu.

5.3 Simulationslauf beenden

Der Simulationslauf endet normalerweise automatisch, sobald alle Aufträge abgearbeitet sind. In PACE erscheint dann ein Hinweisfenster „End of simulation“, das Sie mit **OK** bestätigen müssen.

Sollten Sie ein Abbruchdatum für den Simulationslauf (im Tabellenblatt **Aufträge**) eingegeben haben, so endet der Simulationslauf zu dem angegebenen Datum. Es öffnet sich ein Hinweisfenster mit Details, das Sie (nach dem aufmerksamen Lesen) mit **OK** bestätigen müssen.

Falls Sie einen laufenden Simulationslauf unterbrechen möchten, positionieren Sie den Mauszeiger in einem der Netzfenster (möglicherweise ist der Mauszeiger im Netzfenster

nicht sichtbar – positionieren Sie in diesem Fall den Mauszeiger so, dass er im Netzfenster stehen müsste) und klicken Sie mit der linken Maustaste. Der Simulationslauf sollte daraufhin stoppen. Sollte dies nicht der Fall sein, verschieben Sie den Mauszeiger leicht und klicken Sie wieder auf die linke Maustaste, bis der Simulationslauf stoppt.

Wenn Sie den Simulation fortsetzen wollen, positionieren Sie den Mauszeiger in einem Netzfenster, drücken Sie die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **run** aus und lassen Sie die rechte Maustaste los.

5.4 Auswerten der Excel-Datei

5.4.1 Tabellenblatt Ergebnisse_Aufträge

Nachdem der Simulationslauf beendet ist, können Sie in diesem Tabellenblatt die Ergebnisse der Aufträge ablesen. Im Folgenden sind die Felder des Tabellenblattes von links nach rechts beschrieben:

ID

Hier wird die eindeutige Nummer des Loses angezeigt.

Bitte beachten Sie, dass die Aufträge NICHT in chronologischer Reihenfolge der ID, sondern vielmehr in der Reihenfolge der Fertigstellung angezeigt werden.

Start Auftrag SOLL

Hier wird das Datum angezeigt, das im Tabellenblatt **Aufträge** als Startdatum des Auftrages eingegeben wurde.

Start Auftrag IST

Hier wird das Datum angezeigt, zu dem der Auftrag tatsächlich in die Produktion gegeben wurde.

Fertigstellung SOLL

Hier wird das Datum angezeigt, das im Tabellenblatt **Aufträge** als Datum „Fertig soll (optional)“ eingegeben wurde.

Fertigstellung IST

Hier wird das Datum angezeigt, zu dem der Auftrag fertiggestellt wurde. Das Datum wird **fett rot** markiert, wenn es nach dem Datum „Fertigstellung SOLL“ liegt. Andernfalls wird das Datum **fett grün** markiert.

Durchlaufzeit (Min.)

Hier wird die Zeit in Minuten angezeigt, die der Auftrag tatsächlich vom Start bis zur Fertigstellung benötigt hat.

Rüstzeit: Zeit (Min.)

Hier wird die tatsächliche Rüstzeit des Auftrages in Minuten angezeigt.

Falls Sie die Bearbeitungsstation Gleitschleifmaschine oder externe Bearbeitung gewählt haben, beachten Sie bitte, dass die gesamte Bearbeitungszeit in diesem Feld Rüstzeit: Zeit (Min.) angezeigt wird. Das Feld Bearbeitungszeit: Zeit (Min.) bleibt in diesem Fall frei.

Rüstzeit: in %

Nutzerhandbuch Baukastensystem KMUSimMetall

Hier wird die tatsächliche Rüstzeit des Auftrages in % von der Durchlaufzeit des Auftrages angezeigt.

Bearbeitungszeit: Zeit (Min.)

Hier wird die tatsächliche Bearbeitungszeit des Auftrages in Minuten angezeigt.

Falls Sie die Bearbeitungsstation Gleitschleifmaschine oder externe Bearbeitung gewählt haben, beachten Sie bitte, dass die gesamte Bearbeitungszeit im Feld Rüstzeit: Zeit (Min.) angezeigt wird. Dieses Feld Bearbeitungszeit: Zeit (Min.) bleibt in diesem Fall frei.

Bearbeitungszeit: in %

Hier wird die tatsächliche Bearbeitungszeit des Auftrages in % von der Durchlaufzeit des Auftrages angezeigt.

Wartezeit: Zeit (Min.)

Hier wird die tatsächliche Wartezeit des Auftrages in Minuten angezeigt.

Wartezeit: in %

Hier wird die tatsächliche Wartezeit des Auftrages in % von der Durchlaufzeit des Auftrages angezeigt.

Produktionsplanung

Um Ihre Produktion planen zu können, werden im Folgenden bis zu 10 Stationen angezeigt, auf denen der Auftrag chronologisch nacheinander bearbeitet wurde.

Produktionsplanung: Station

Hier wird die jeweilige Station angezeigt.

Produktionsplanung: Start auf Station

Hier wird das Startdatum des Auftrages auf der betreffenden Station angezeigt.

5.4.2 Tabellenblatt Ergebnisse_Stationen

Nachdem der Simulationslauf beendet ist, können Sie in diesem Tabellenblatt die Ergebnisse der Stationen ablesen. Für jede Kategorie von Bearbeitungsstationen wird eine Zeile aufgelistet.

Im Folgenden sind die Felder des Tabellenblattes von links nach rechts beschrieben:

Simulierte Zeit

Hier wird die simulierte Zeit in Minuten angezeigt.

Kategorie Bearbeitungsstation

Hier wird die jeweilige Bezeichnung der Kategorie von Bearbeitungsstationen angezeigt. Die Bezeichnungen der Bearbeitungsstationen sind aus systemtechnischen Gründen abgekürzt angegeben.

Betriebszeit Stationen

Hier wird die tatsächliche Betriebszeit aller Stationen der betreffenden Kategorie angezeigt.

Nutzerhandbuch Baukastensystem KMUSimMetall

Beispiel: Die simulierte Zeit beträgt 600 Minuten, es gibt 3 Bearbeitungsstationen der Kategorie CNC-Drehmaschinen. Die Betriebszeit Stationen wird mit 1800 Minuten (3 Stationen x 600 Minuten) angezeigt.

Rüstzeit: Zeit (Min.)

Hier wird die tatsächliche Rüstzeit aller Stationen der betreffenden Kategorie zusammengesetzt in Minuten angezeigt.

Beispiel: Es gibt 3 Bearbeitungsstationen der Kategorie CNC-Drehmaschinen. CNC-Drehmaschine 1 hatte eine Rüstzeit von 100 Minuten, CNC-Drehmaschine 2 hatte eine Rüstzeit von 70 Minuten, CNC-Drehmaschine 3 hatte eine Rüstzeit von 30 Minuten. Die hier angezeigte Rüstzeit: Zeit (Min.) wird mit 200 Minuten (100 Minuten + 70 Minuten + 30 Minuten) angezeigt.

Da bei den Gleitschleifmaschinen und der externen Bearbeitung die komplette Bearbeitungszeit unter der Rüstzeit eingegeben wird, entspricht bei diesen Kategorien die hier angezeigte Rüstzeit tatsächlich der Bearbeitungszeit.

Rüstzeit: in %

Hier wird die tatsächliche Rüstzeit aller Stationen der betreffenden Kategorie zusammengesetzt in % (aufgerundet) der **simulierten Zeit** angezeigt.

*Beispiel: Es gibt 3 Bearbeitungsstationen der Kategorie CNC-Drehmaschinen, die zusammen eine Rüstzeit von 200 Minuten (100 Minuten + 70 Minuten + 30 Minuten) benötigen. Die **simulierte Zeit** betrug 2000 Minuten. Die Rüstzeit: in % wird mit 10% (200 Minuten entsprechen 10% von 2000 Minuten) angezeigt.*

Bearbeitungszeit: Zeit (Min.)

Hier wird die tatsächliche Bearbeitungszeit aller Stationen der betreffenden Kategorie zusammengerechnet in Minuten angezeigt.

Beispiel: Es gibt 3 Bearbeitungsstationen der Kategorie CNC-Drehmaschinen. CNC-Drehmaschine 1 hatte eine Bearbeitungszeit von 50 Minuten, CNC-Drehmaschine 2 hatte eine Bearbeitungszeit von 35 Minuten, CNC-Drehmaschine 3 hatte eine Bearbeitungszeit von 15 Minuten. Die hier angezeigte Bearbeitungszeit: Zeit (Min.) wird mit 100 Minuten (50 Minuten + 35 Minuten + 15 Minuten) angezeigt.

Hinweis: Da bei den Gleitschleifmaschinen und der externen Bearbeitung die komplette Bearbeitungszeit unter der Rüstzeit eingegeben wird, bleibt bei diesen Kategorien die Bearbeitungszeit leer; siehe auch Spalte **Rüstzeit: Zeit (Min.)**.

Bearbeitungszeit: in %

Hier wird die tatsächliche Bearbeitungszeit aller Stationen der betreffenden Kategorie zusammengerechnet in % (aufgerundet) der **simulierten Zeit** angezeigt.

*Beispiel: Es gibt 3 Bearbeitungsstationen der Kategorie CNC-Drehmaschinen, die zusammen eine Bearbeitungszeit von 100 Minuten (50 Minuten + 35 Minuten + 15 Minuten) benötigen. Die **simulierte Zeit** betrug 1000 Minuten. Die Bearbeitungszeit: in % wird mit 10% (100 Minuten entsprechen 10% von 1000 Minuten) angezeigt.*

Wartezeit: Zeit (Min.)

Hier wird die tatsächliche Wartezeit aller Stationen der betreffenden Kategorie zusammengesamt in Minuten angezeigt.

Beispiel: Es gibt 2 Bearbeitungsstationen der Kategorie Gleitschleifmaschine. Gleitschleifmaschine 1 hatte eine Wartezeit von 600 Minuten, Gleitschleifmaschine 2 hatte eine Wartezeit von 100 Minuten. Die hier angezeigte Wartezeit: Zeit (Min.) wird mit 610 Minuten (600 Minuten + 100 Minuten) angezeigt.

Wartezeit: in %

Hier wird die tatsächliche Wartezeit aller Stationen der betreffenden Kategorie zusammengesamt in % (aufgerundet) der **simulierten Zeit** angezeigt.

*Beispiel: Es gibt 2 Bearbeitungsstationen der Kategorie Gleitschleifmaschinen, die zusammen eine Bearbeitungszeit von 700 Minuten (600 Minuten + 100 Minuten) benötigten. Die **simulierte Zeit** betrug 7000 Minuten. Die Wartezeit: in % wird mit 10% (700 Minuten entsprechen 10% von 7000 Minuten) angezeigt.*

Anzahl

Hier wird die in PACE eingefügte Anzahl von Bearbeitungsstationen der betreffenden Kategorie angezeigt.

5.5 Speichern der Ergebnisse des Simulationslaufes

Wenn Sie die Ergebnisse des Simulationslaufes in Excel speichern wollen, so ist das möglich, indem Sie eine Kopie Ihrer geänderten Excel-Datei unter einem anderen Namen speichern (die im **Baukastensystem KMUSimMetall** enthaltene Excel-Datei **Befüllung.xlsx** ist geschützt, sie kann also nicht unter diesem Namen abgespeichert werden; dies soll sicherstellen, dass Ihnen die Originaldatei immer zur Verfügung steht).

Bitte beachten Sie, dass Sie die von Ihnen erstellte Kopie zunächst nicht für neue Simulationsläufe nutzen können, da die Excel-Datei ZWINGEND den Namen **Befüllung** und das Dateiformat **xlsx** tragen muss. Sie können dieses Problem lösen, indem Sie Ihre unter einem anderen Dateinamen abgespeicherte (und geschlossene) Kopie der Excel-Datei anschließend im Windows-Explorer in den Namen **Befüllung** im Dateiformat **xlsx** umbenennen. Um eine Verwechslung mit der Originaldatei zu vermeiden, ist anzuraten, diese veränderte Excel-Datei in einen separaten Dateiordner, der beispielsweise als **Eigene Excel-Dateien** bezeichnet wird, abzulegen.

5.6 Simulationslauf im Hintergrund

Für den Fall dass Sie viele Aufträge mit vielen Werkstücken in die Excel-Datei eingepflegt haben, dauert der Simulationslauf unter Umständen relativ lange (ein Simulationslauf mit 100 Aufträgen zu je 50000 Teilen unter der Nutzung verschiedener Kategorien von Bearbeitungsstationen dauert etwa 4 Stunden). Um dies zu beschleunigen, können Sie die Simulation auch „im Hintergrund“ laufen lassen. Gehen Sie dazu zum Start des Simulationslaufes vor, wie im Kapitel 5.2 auf Seite 16 beschrieben, wählen Sie aber statt des Menüpunktes **initialize + run** den Menüpunkt **initialize + background run** aus. Der Simulationslauf wird dann nicht animiert dargestellt, dauert aber dafür nur etwa halb so lang.

6 Anpassen an Ihr Unternehmen

6.1 Jedes KMU ist anders

Im Basisbaustein ist jede mögliche Kategorie von Bearbeitungsstationen mit jeweils einer Bearbeitungsstation enthalten. Sollten in Ihrem Unternehmen einzelne Kategorien von Bearbeitungsstationen gar nicht oder mit mehr als einer Bearbeitungsstation vorhanden sein, müssen Sie das Simulationsmodell an Ihr Unternehmen anpassen.

6.2 Löschen von Kategorien von Bearbeitungsstationen

Sollte eine Kategorie von Bearbeitungsstationen in Ihrem Unternehmen gar nicht vorhanden sein (sie haben z.B. keine manuellen Schleifmaschinen in Ihrer Werkstatt), können Sie diese Kategorie aus Ihrem Simulationsmodell löschen (in Excel existiert die Kategorie aus systemtechnischen Gründen weiterhin).

Gehen Sie dazu wie folgt vor:

Stellen Sie sicher, dass sich in PACE das Netzfenster Maschinenpark im edit-Modus befindet (dies kann links am unteren Rand des betreffenden Netzfensters umgeschaltet werden). Klicken Sie dann im Netzfenster Maschinenpark das Quadrat der betreffenden Kategorie von Bearbeitungsstationen mit der linken Maustaste an. Anschließend drücken Sie die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **delete** aus und lassen Sie die rechte Maustaste los. Die Sicherheitsabfrage von PACE beantworten Sie mit **yes**.

Die betreffende Kategorie sollte nun aus dem Netzfenster Maschinenpark verschwunden sein.

6.3 Einfügen von Kategorien von Bearbeitungsstationen

Bitte fügen Sie jede **Kategorie von Bearbeitungsstationen** nur 1x in Ihr Simulationsmodell ein (Sie können natürlich mehrere Bearbeitungsstationen in eine solche Kategorie einfügen, siehe im nächsten Kapitel 6.4 auf Seite 24). Sollten Sie jedoch eine **Kategorie von Bearbeitungsstationen** (z.B. die Kategorie CNC-Drehmaschinen) mehrfach in Ihren Maschinenpark einfügen, kann dies zu Fehlermeldungen in PACE und zu falschen Ergebnissen in Excel führen.

Sollten Sie eine Kategorie von Bearbeitungsstationen gelöscht haben und nachträglich wieder einfügen wollen, gehen Sie wie folgt vor:

Stellen Sie sicher, dass sich in PACE das Netzfenster Maschinenpark im edit-Modus befindet (dies kann links am unteren Rand des betreffenden Netzfensters umgeschaltet werden, siehe auch Abbildung 18 auf Seite 17).

Positionieren Sie den Mauszeiger in dem Netzfenster Maschinenpark, drücken Sie die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **restore module** aus und lassen Sie die rechte Maustaste los. Es öffnet sich ein Windows-Explorer-Fenster, in dem Sie bitte aus dem Dateiordner **Kategorien von Bearbeitungsstationen** (Unterordner des Dateiordners **PACE-Bausteine**) den gewünschten PACE-Baustein auswählen. Mit **Öffnen** bestätigen.

Nutzerhandbuch Baukastensystem KMUSimMetall

Es erscheint ein schwarzes Quadrat, das „an Ihrem Mauszeiger klebt“. Bewegen Sie den Mauszeiger an eine passende Stelle im Netzfenster Maschinenpark und positionieren Sie den Baustein durch Klicken mit der linken Maustaste (Sie können den Baustein auch nachträglich noch durch Anklicken mit der linken Maustaste und anschließendes Ziehen verschieben). Es erscheint ein Quadrat, das mit der ausgewählten Kategorie (z.B. CNC-Drehmaschinen) beschriftet und mit einem weißen Kreis „Zwischenlager“ (schwarzer Rand) verbunden ist.

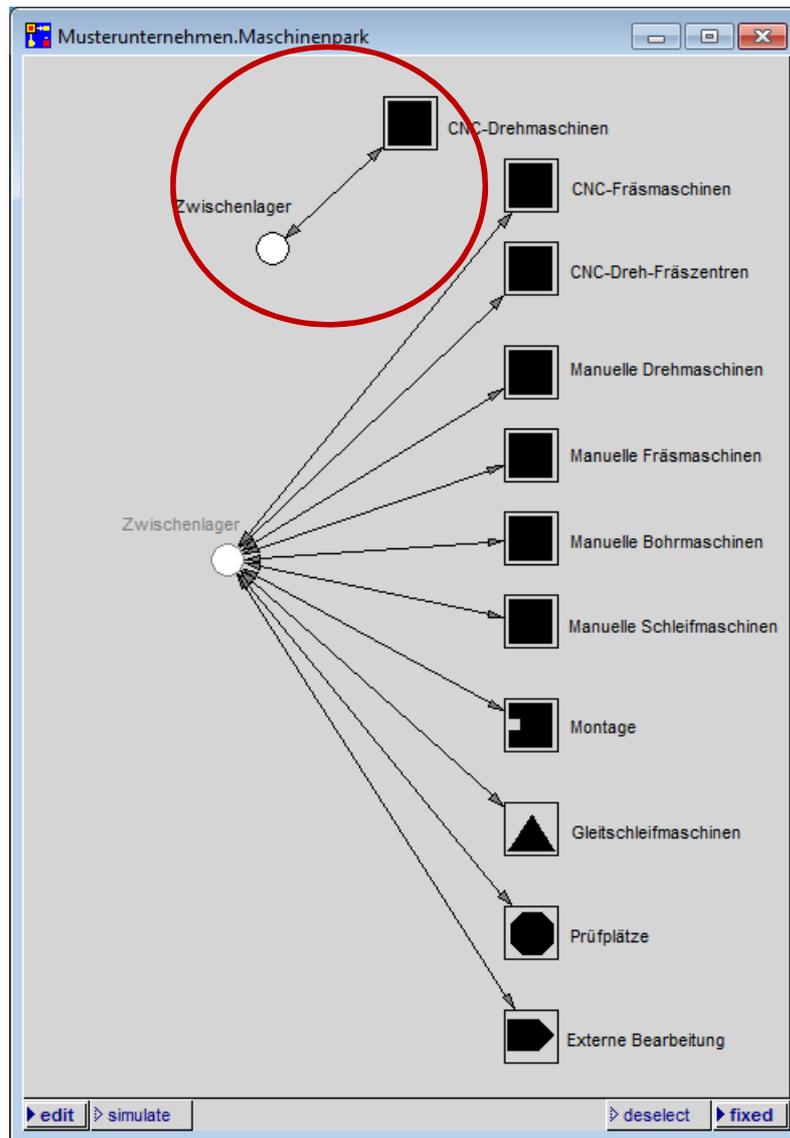


Abbildung 19: Neuer Baustein eingefügt

Nun müssen Sie diesen Baustein noch an Ihr bereits vorhandenes Zwischenlager (grauer Rand) anbinden. Klicken Sie dafür auf den Pfeil, der den neu eingefügten Baustein mit dem weißen Kreis „Zwischenlager“ (schwarzer Rand) verbindet. Es erscheint etwa in der Mitte des Verbindungspfeiles ein kleines schwarzes Quadrat. Klicken Sie dieses schwarze Quadrat mit der linken Maustaste an und ziehen Sie den Verbindungspfeil mit der linken Maustaste (es erscheint ein Kreuz anstelle des Mauszeigers) auf den bereits vorhandenen weißen Kreis „Zwischenlager“ (grauer Rand). Dies wiederholen Sie mit dem zweiten Pfeil. Das Ergebnis sollte dann so aussehen:

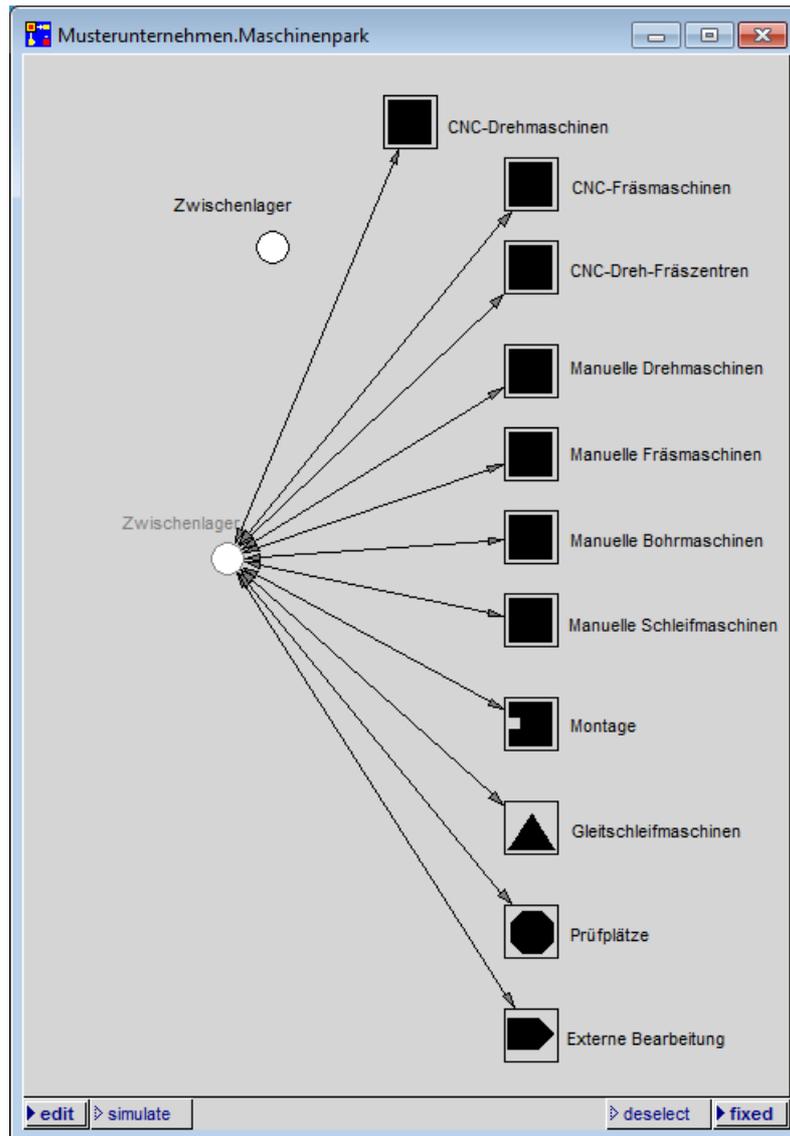


Abbildung 20: Neuer Baustein angebunden

Nun können Sie abschließend den Kreis „Zwischenlager“ (schwarzer Rand) löschen. Klicken Sie dazu den Kreis „Zwischenlager“ (schwarzer Rand) mit der linken Maustaste an. Anschließend drücken Sie die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **delete** aus und lassen Sie die rechte Maustaste los. Die Sicherheitsabfrage von PACE beantworten Sie mit **yes**.

6.4 Einfügen von Bearbeitungsstationen

Sollte in Ihrem Unternehmen von einer Kategorie von Bearbeitungsstationen mehr als eine Bearbeitungsstation vorhanden sein, müssen Sie diese zusätzlich in die jeweilige Kategorie einfügen. Dazu gehen Sie wie folgt vor:

Stellen Sie sicher, dass sich in PACE das Netzfenster Maschinenpark im edit-Modus befindet (dies kann links am unteren Rand des betreffenden Netzfensters umgeschaltet werden, siehe auch Abbildung 18 auf Seite 17).

Klicken Sie in dem Netzfenster Maschinenpark das Quadrat der entsprechenden Kategorie, die Sie anpassen möchten, mit der linken Maustaste an. Drücken Sie dann die rechte Maus-

Nutzerhandbuch Baukastensystem KMUSimMetall

taste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **subnet** aus und lassen Sie die rechte Maustaste los. Es öffnet sich ein Fenster, das „an Ihrem Mauszeiger klebt“. Bewegen Sie den Mauszeiger an einen freien Platz auf Ihrem Bildschirm und positionieren Sie das neue Fenster durch Drücken der linken Maustaste. Ihr Bildschirm sollte nun so aussehen:

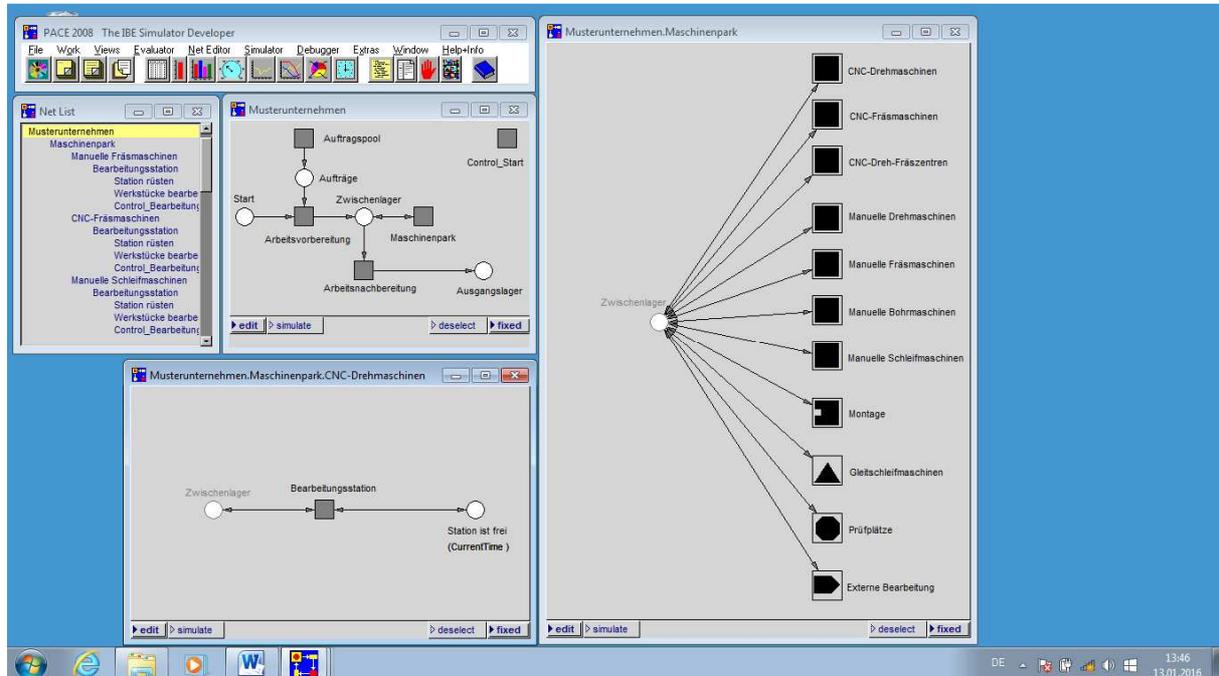


Abbildung 21: Bildschirmansicht (2)

Anschließend positionieren Sie den Mauszeiger in dem neuen Netzfenster, drücken Sie die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **restore Module** aus und lassen Sie die rechte Maustaste los. Es öffnet sich ein Windows-Explorer-Fenster, in dem Sie bitte aus dem Dateiordner **PACE-Bausteine** die Datei **Bearbeitungsstation.sub** auswählen. Mit **Öffnen** bestätigen.

Es erscheint ein schwarzes Quadrat, das „an Ihrem Mauszeiger klebt“. Bewegen Sie den Mauszeiger an eine freie Stelle des neuen Netzfensters und positionieren Sie die Bearbeitungsstation durch Klicken mit der linken Maustaste. Es erscheint ein graues Quadrat, das als Bearbeitungsstation beschriftet und mit einem weißen Kreis „Zwischenlager“ (schwarzer Rand) sowie einem weißen Kreis „Station ist frei“ (schwarzer Rand) verbunden ist.

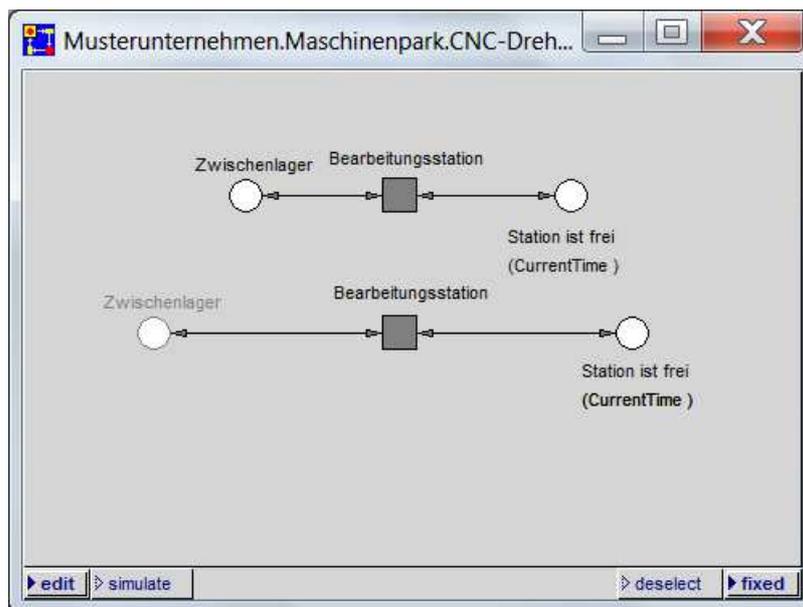


Abbildung 22: Neue Bearbeitungsstation eingefügt

Nun müssen Sie die neue Bearbeitungsstation noch an ihr bereits vorhandenes Zwischenlager (grauer Rand) anbinden. Klicken Sie dafür auf den Pfeil, der den neu eingefügten Baustein mit dem weißen Kreis „Zwischenlager“ (schwarzer Rand) verbindet. Es erscheint etwa in der Mitte des Verbindungspfeiles ein kleines schwarzes Quadrat. Klicken Sie dieses schwarze Quadrat mit der linken Maustaste an und ziehen Sie den Verbindungspfeil mit der linken Maustaste (es erscheint ein Kreuz anstelle des Mauszeigers) auf den bereits vorhandenen weißen Kreis „Zwischenlager“ (grauer Rand). Dies wiederholen Sie mit dem zweiten Pfeil. Das Ergebnis sollte dann so aussehen:

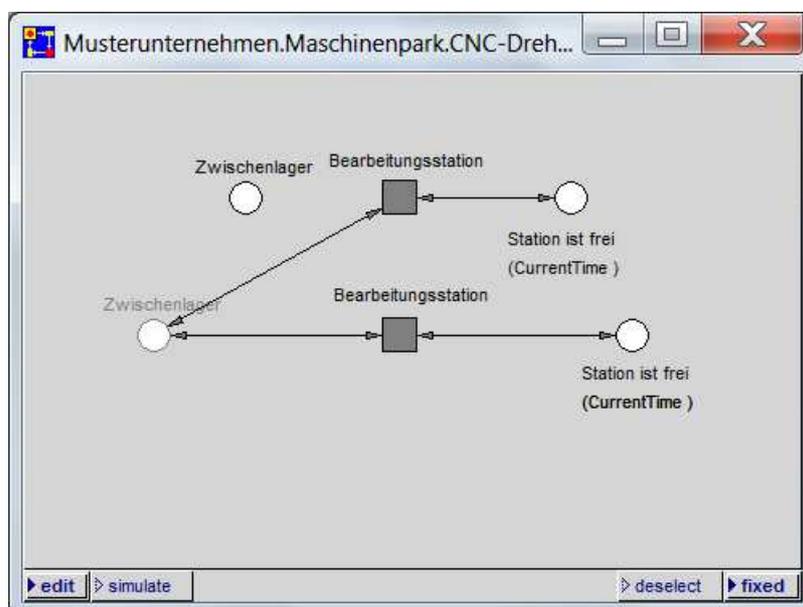


Abbildung 23: Neue Bearbeitungsstation angebunden

Nun können Sie den Kreis „Zwischenlager“ (schwarzer Rand) löschen. Klicken Sie dazu den Kreis „Zwischenlager“ (schwarzer Rand) mit der linken Maustaste an. Anschließend drücken Sie die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter

Nutzerhandbuch Baukastensystem KMUSimMetall

Maustaste den erscheinenden Menüpunkt **delete** aus und lassen Sie die rechte Maustaste los. Die Sicherheitsabfrage von PACE beantworten Sie mit **yes**.

Abschließend schließen Sie das neue Netzfenster. Ihr Bildschirm sollte nun wieder so aussehen:

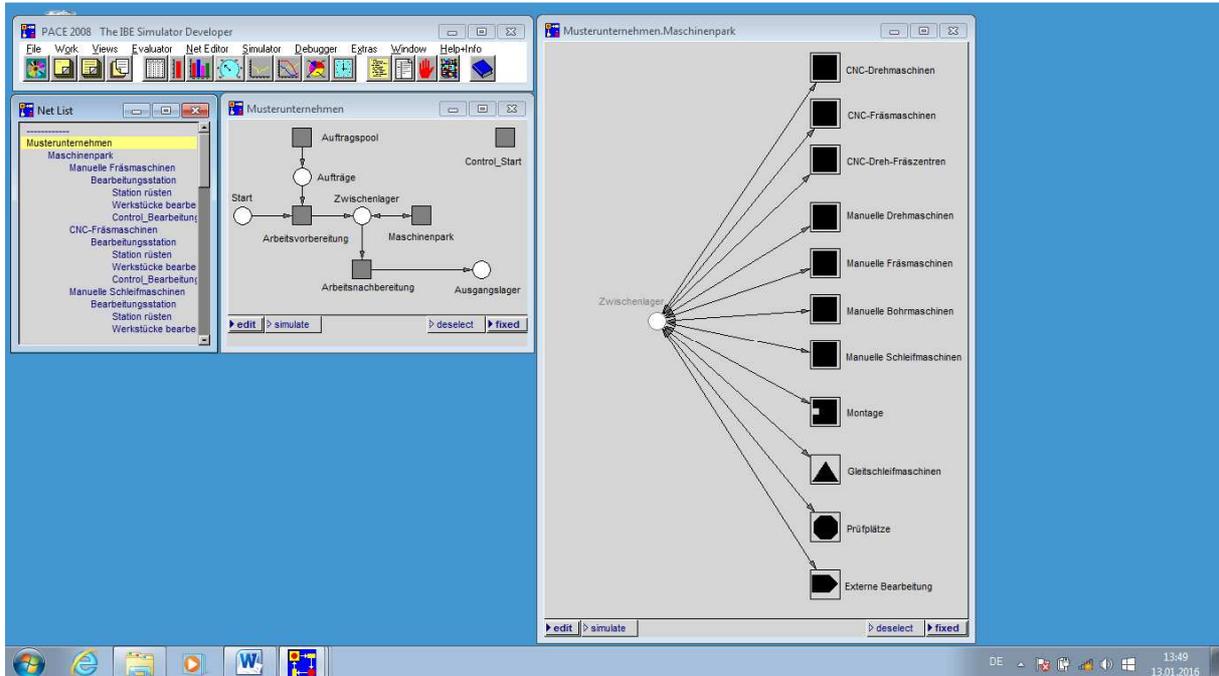


Abbildung 24: Bildschirmansicht (3)

Sie können das Ergebnis kontrollieren, indem Sie einen Simulationslauf ausführen und im Tabellenblatt **Ergebnisse_Stationen** der Excel-Datei die betreffende Kategorie von Bearbeitungsstationen unter dem Punkt „Anzahl“ kontrollieren.

6.5 Löschen von Bearbeitungsstationen

Sollte sich in Ihrem Unternehmen die Anzahl von Bearbeitungsstationen einer Kategorie ändern, müssen Sie Ihr Simulationsmodell anpassen. Gehen Sie dazu wie folgt vor:

Stellen Sie sicher, dass sich in PACE das Netzfenster Maschinenpark im edit-Modus befindet (dies kann links am unteren Rand des betreffenden Netzfensters umgeschaltet werden, siehe auch Abbildung 18 auf Seite 17).

Klicken Sie in dem Netzfenster Maschinenpark das Quadrat der entsprechenden Kategorie von Bearbeitungsstationen, die Sie anpassen möchten, mit der linken Maustaste an. Drücken Sie dann die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Punkt **subnet** aus und lassen Sie die rechte Maustaste los. Es öffnet sich ein Fenster, das „an Ihrem Mauszeiger klebt“. Bewegen Sie den Mauszeiger an einen freien Platz auf Ihrem Bildschirm und positionieren Sie das neue Netzfenster durch Drücken der linken Maustaste. Ziehen Sie das Netzfenster gegebenenfalls größer, so dass alle Elemente gut zu sehen sind. Ihr Bildschirm sollte nun so aussehen:

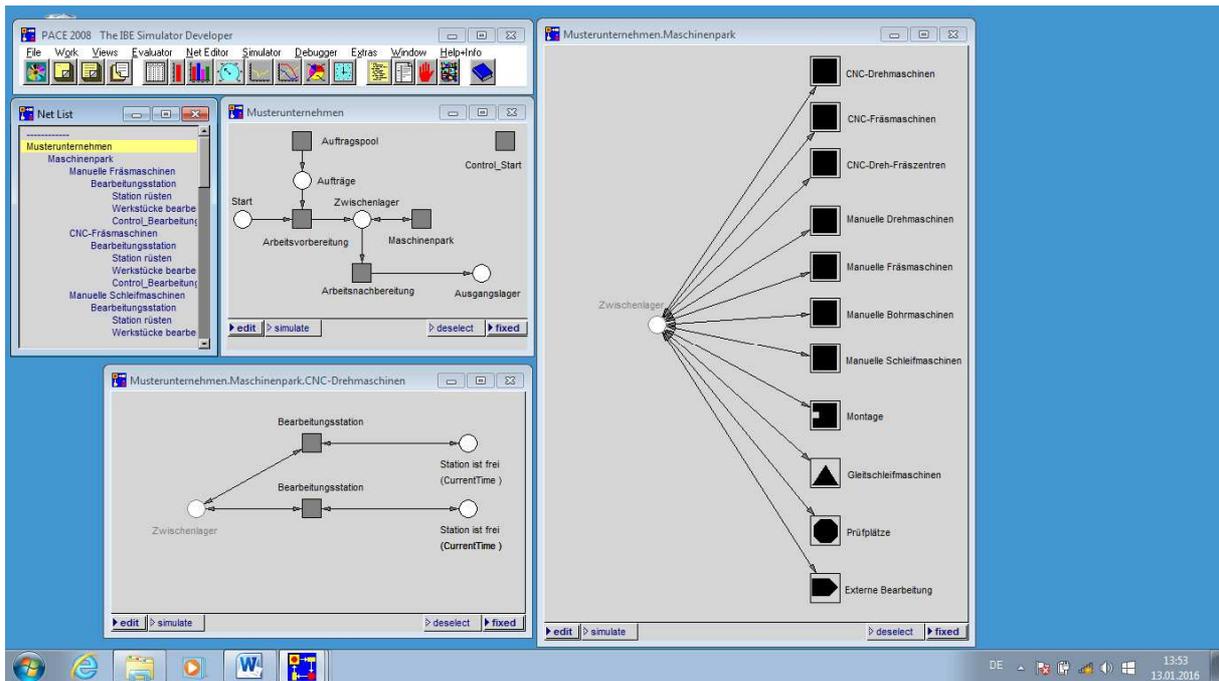


Abbildung 25: Bildschirmansicht (4)

Klicken Sie dann in dem neuen Netzfenster das Quadrat einer Bearbeitungsstation mit der linken Maustaste an. Drücken Sie dann die SHIFT-Taste Ihrer Tastatur, HALTEN SIE DIESE GEDRÜCKT und klicken Sie mit der linken Maustaste auch den verbundenen weißen Kreis „Station ist frei“ im neuen Netzfenster an. Lassen Sie die SHIFT-Taste Ihrer Tastatur los und drücken Sie die rechte Maustaste UND HALTEN SIE DIESE GEDRÜCKT. Wählen Sie mit gedrückter Maustaste den erscheinenden Menüpunkt **delete** aus und lassen Sie die rechte Maustaste los. Die Sicherheitsabfrage von PACE beantworten Sie mit **yes**.

Die betreffende Bearbeitungsstation sollte nun verschwunden sein. Nun können Sie das neue Netzfenster schließen.

Sie können das Ergebnis kontrollieren, indem Sie einen Simulationslauf ausführen und im Tabellenblatt **Ergebnisse Stationen** der Excel-Datei die betreffende Kategorie von Bearbeitungsstationen unter dem Punkt „Anzahl“ kontrollieren.

6.6 Speichern Ihres Simulationsmodelles

Um das an Ihr Unternehmen angepasste Simulationsmodell zu sichern, müssen Sie es abspeichern. Am praktikabelsten ist das Abspeichern als PACE-Image.

Dazu rufen Sie in der PACE-Bedienleiste (beschriftet mit **PACE 2008 The IBE Simulator Developer**) bitte den Menüpunkt **File** auf. Wählen Sie mit der linken Maustaste den Menüpunkt **store image** aus. Es öffnet sich ein Windows-Explorer-Fenster. Wählen Sie am besten den Dateiordner **KMUSimMetall** aus und benennen Sie das Simulationsmodell mit dem Namen Ihres Unternehmens. Bestätigen Sie mit **Speichern**.

Ihr so gespeichertes Simulationsmodell sollte sich zukünftig per Doppelklick auf die soeben erstellte Datei direkt aus dem Windows-Explorer öffnen lassen.

7 Schließen der Dateien

7.1 Schließen von PACE

Um das Simulationsmodell in PACE zu schließen, rufen Sie in der PACE-Bedienleiste (beschriftet mit **PACE 2008 The IBE Simulator Developer**) bitte den Menüpunkt **File** auf. Wählen Sie mit der linken Maustaste den Menüpunkt **leave PACE** aus. Die anschließende Sicherheitsabfrage beantworten Sie mit **no**.

PACE sollte sich nun mit all seinen Netzfenstern schließen.

7.2 Schließen von Excel

Sie können die Excel-Datei wie von Ihrer Office-Software gewohnt schließen. Wenn Sie Eingaben und Ergebnisse sichern wollen, lesen Sie bitte unter Kapitel 5.5 auf Seite 21 nach.