



Handelshochschule
Stockholm



OTTO VON GUERICKE
UNIVERSITÄT
MAGDEBURG

INF

FAKULTÄT FÜR
INFORMATIK

Simulation, Animation und Simulationsprojekt

Thema 1:
Modellierung eines Bediensystems

Dr. Henry Herper – Otto-von-Guericke-Universität Magdeburg - SS 2019

Phasen einer Simulationsstudie

Phase	ausgewählte Informatikinhalt
Auswahl und Beschreibung des Bediensystems	<ul style="list-style-type: none">• Erarbeitung des System und Modellbegriffs• Erfassung und Aufbereitung empirischer Daten
Entwicklung des abstrakten Modells	<ul style="list-style-type: none">• Erlernen der Methoden von Abstraktion und Reduktion• Arbeit mit formalen Beschreibungsmöglichkeiten von Systemen
Implementierung des Computermodells	<ul style="list-style-type: none">• Erlernen einer Simulationssprache bzw. Anwendung einer bekannten Programmiersprache auf ein komplexes System• Erlernen grundlegender Validierungstechniken

Phasen einer Simulationsstudie

Phase	ausgewählte Informatikinhalt
Durchführung von Modellexperimenten	<ul style="list-style-type: none">• Erarbeiten von Experimentierstrategien• Bewerten der Ergebnisse und Ableiten von Eingabedaten für weitere Experimente
Aufbereitung und Präsentation der Ergebnisse	<ul style="list-style-type: none">• Erlernen von Visualisierungs- und Präsentationstechniken• kritische Betrachtung der Ergebnisse von Berechnungen• Erkenntnisgewinn durch Simulationsstudien

Finden eines geeigneten Systems und einer geeigneten Aufgabenstellung

Auswahl und Beschreibung eines Bediensystems

- Auswahl eines **Bediensystems aus dem Erfahrungshorizont der Schüler**, zum Beispiel Dienstleistungsbetrieb, Supermarkt, Tankstelle
- Erstellen einer geeigneten Aufgabenstellung, die mit dem Simulationsmodell beantwortet und anschließend **validiert** werden kann
- Modellerstellung und Durchführung der Modellexperimente muss mit den zur Verfügung stehenden Mitteln in der zur Verfügung stehenden Zeit möglich sein

Beispiel: Reales System - Tankstelle



/Quelle: <http://maps.live.com/>

Ziele der Simulationsstudie

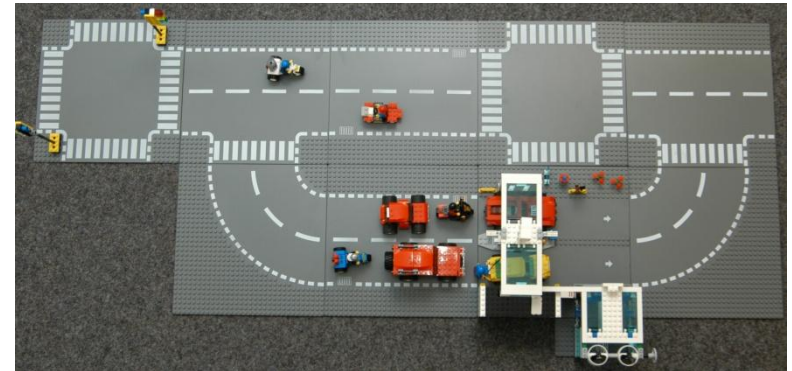
- Wie viele Zapfsäulen sind notwendig, damit die Wartezeit der Fahrzeuge in 90% der Fälle kleiner als 10 Minuten ist?
- Welche Kapazität muss der Wartebereich für Fahrzeuge haben?
- Ist es zweckmäßig, gesonderte Stellplätze für Kunden zu haben, die nur im Tankstellenshop einkaufen möchten?
- Wie viele Kassen sind notwendig?
- Verständnis der Abläufe im System.

Voraussetzung der Modellierung: Abstraktion

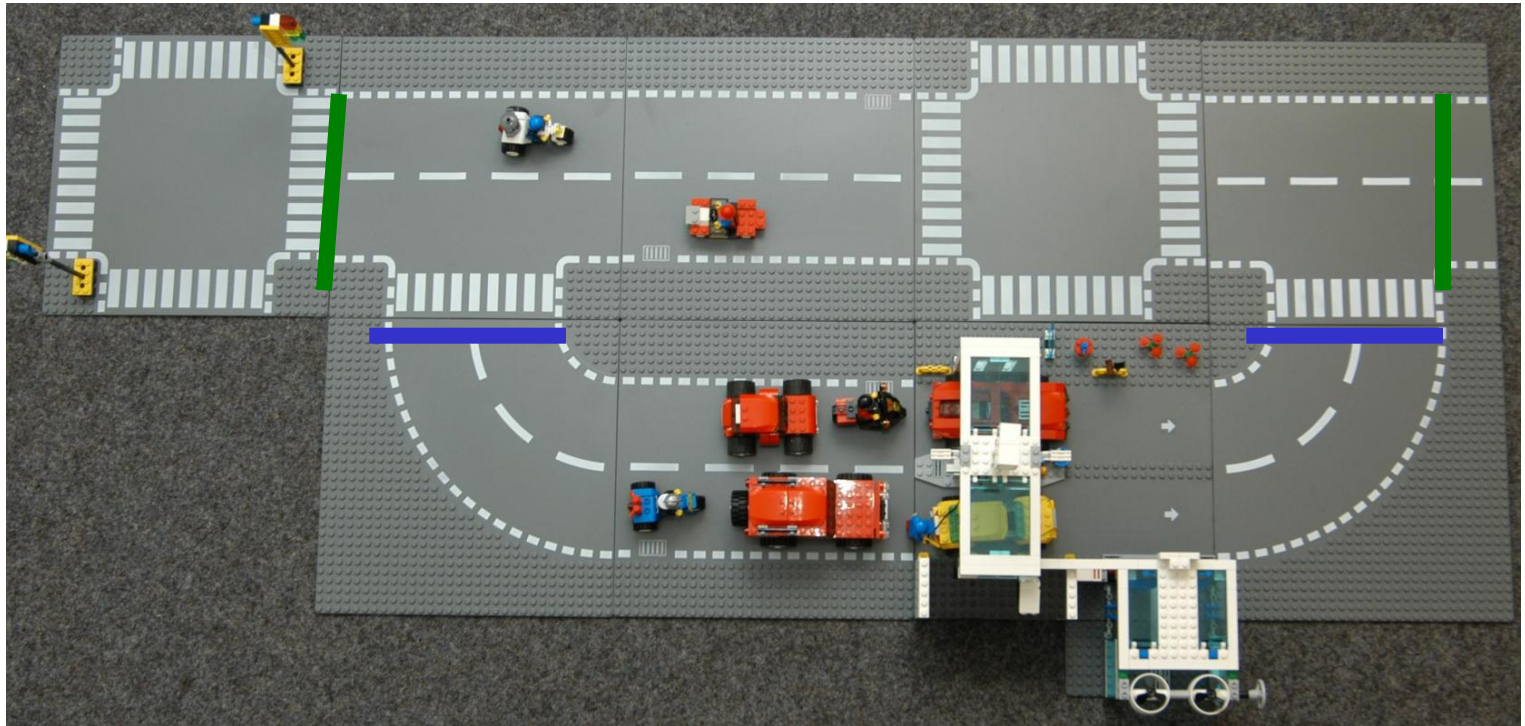


reale Tankstelle

physisches Modell einer
Tankstelle



Festlegen des Systemgrenzen - Umweltschnittstellen



1. Möglichkeit: Straßenabschnitt einbeziehen
2. Möglichkeit: Tankstelleneinfahrt und –ausfahrt
3. ...

Prozessbegriff

Prozessbegriff:

Unter einem **Prozess** (lat. procedere = voranschreiten; PPP: processus) versteht man eine definierte oder wahrscheinliche Aufeinanderfolge von Zuständen eines Systems in Abhängigkeit von den Vorbedingungen und den äußeren Einflüssen. Der Ablauf eines Prozesses kann vorgegeben sein, meist aber auch eigenständig gestaltet werden.

Deterministische Prozesse sind solche, bei denen jeder Zustand eindeutig aus dem ihm vorangegangenen Zustand hervorgeht.

Stochastische Prozesse hingegen haben das Besondere an sich, dass die Zustände eines Systems nur mit einer gewissen Wahrscheinlichkeit aus den ihnen vorangegangenen Zuständen folgen.

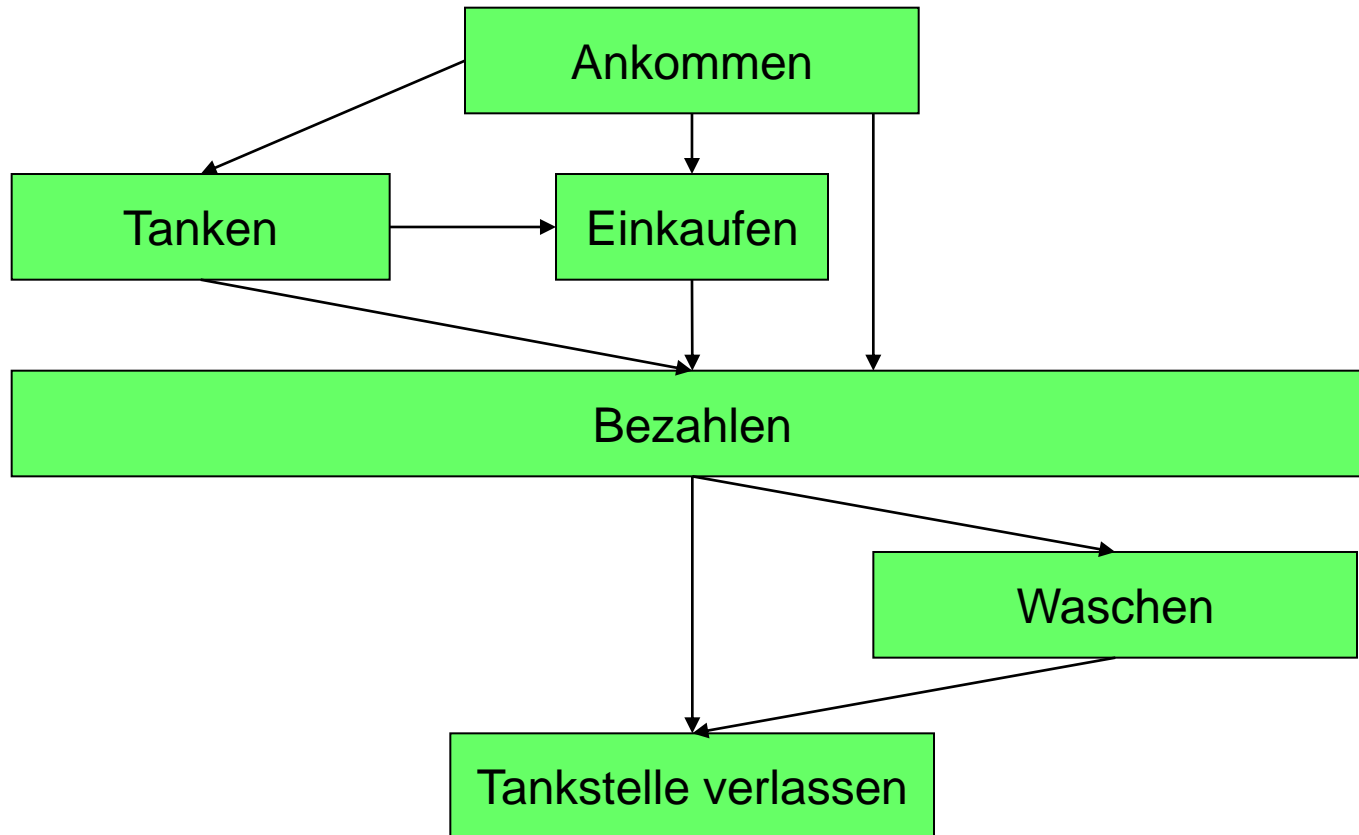
/Quelle: <http://www.wikipedia.de//>

Prozesse im realen System

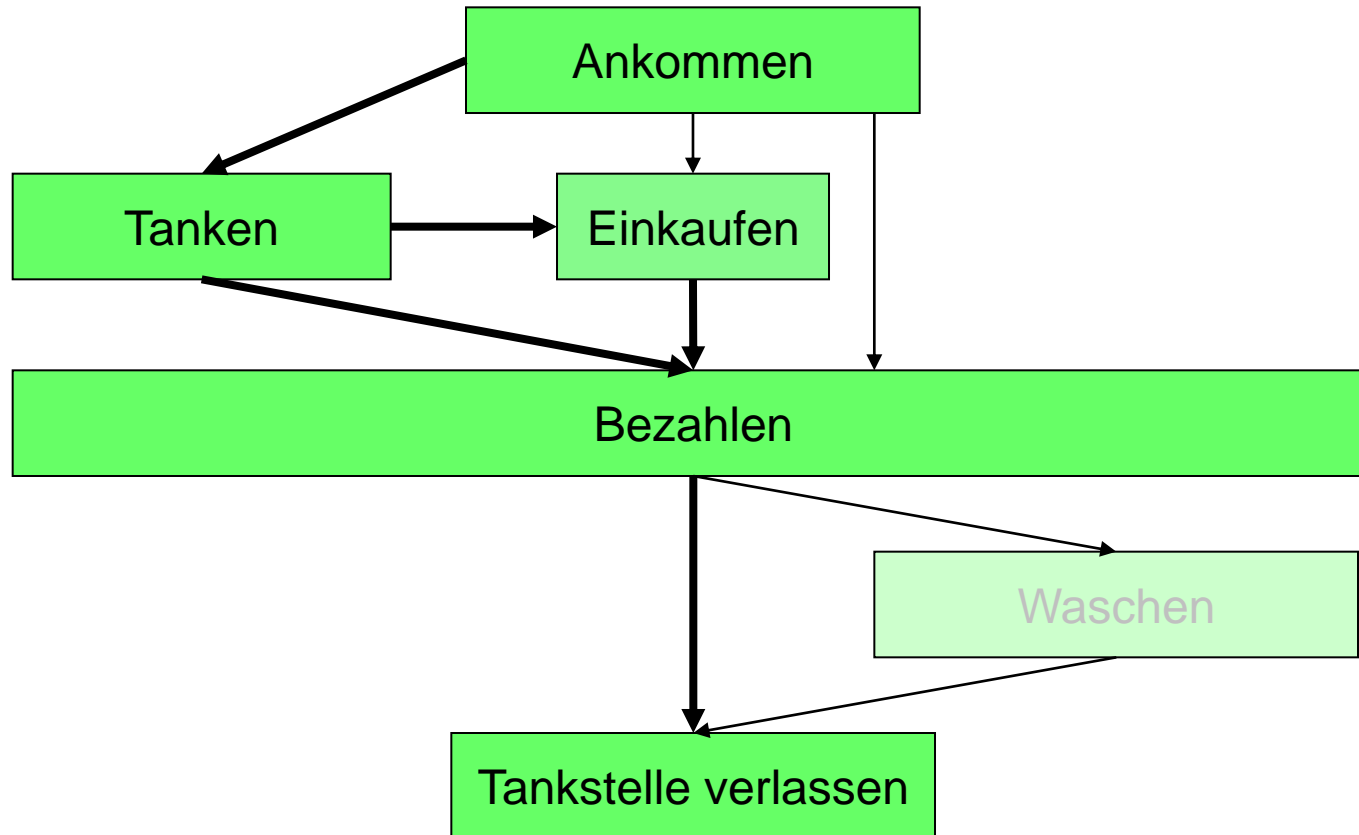
Hauptprozesse:

- Fahrt von der Einfahrt zur Zapfsäule bzw. zum Parkplatz für den Shop
- Tanken
- Einkaufen im Shop
- Bezahlen
- Reinigen der Scheiben
- Kaufen einer Öldose am Automaten
- Waschen des Autos
- Fahrt von der Zapfsäule, Parkplatz oder Waschanlage zum Ausgang

Handlungsfolgen im System



Im Rollenspiel nachgebildete Handlungsfolge



Klassen im System

- PKW
- Motorrad
- Fahrer (Person, die tankt bzw. einkauft)
- Zapfsäule
- Tankwart
- Kassierer
- Waschanlage
- Öldosenautomat

Klasse PKW

reales Fahrzeug

/Quelle:

<http://www.wikipedia.de//>



Lego-Modell

Fahrzeug
+ Typkennzeichen : int - Nummer : int + Tankinhalt : int - Streckenfahrzeit : float - Handlungsplan : byte
+ setTypkennzeichen(TK : int) : void + getTypkennzeichen() : int + setNummer(NR : int) : void + getNummer() : int + setTank(Volumen : int) : void + getTank() : int + setZeit(Fahrzeit : float) : void + getZeit() : float + setHandlungsplan(aktion : byte) : void + getHandlungsplan() : byte

UML-Klasse

Animations-
Klasse



Klasse Bike

reales Fahrzeug

/Quelle:
<http://www.wikipedia.de//>

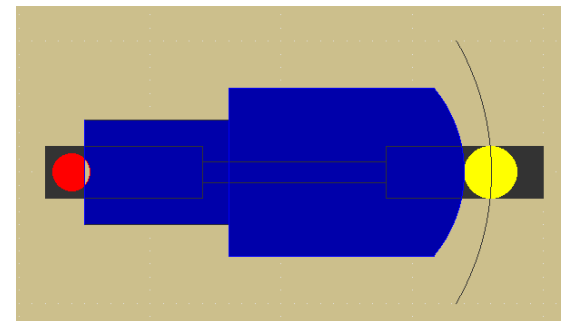
Lego-Modell



Fahrzeug
+ Typkennzeichen : int
- Nummer : int
+ Tankinhalt : int
- Streckenfahrzeit : float
- Handlungsplan : byte
+ setTypkennzeichen(TK : int) : void
+ getTypkennzeichen() : int
+ setNummer(NR : int) : void
+ getNummer() : int
+ setTank(Volumen : int) : void
+ getTank() : int
+ setZeit(Fahrzeit : float) : void
+ getZeit() : float
+ setHandlungsplan(aktion : byte) : void
+ getHandlungsplan() : byte

UML-Klasse

Animations-
Klasse



Anwendungsfall: Tanken

Ziel: Fahrzeug mit Treibstoff zu befüllen

Anwendungskontext: Betanken eines Fahrzeuges an einer Tankstelle durch Kraftfahrer

Bereich: Zum System gehört die Zapfsäule und der Eimer mit Waschwasser, nicht zum System gehören Shop, Kasse, Waschanlage, Straße

Primärer Akteur: Kraftfahrer

Sekundärer Akteur: Bediener der Tankstellenanlage

Betroffene (Stakeholder): andere Tankkunden

Vorbedingungen: Fahrzeug benötigt Kraftstoff, benötigter Kraftstoff ist vorhanden

Nachbedingung im Erfolgsfall: Fahrzeug bekommt benötigten Kraftstoff



Anwendungsfall: Tanken

Nachbedingung in Fehlerfällen:

- Wenn kein Treibstoff vorhanden, dann Tanken nicht möglich.
- Wenn sich der Tankdeckel nicht öffnen lässt, dann Tanken nicht möglich.

Auslöser: Fahrzeug erreicht Zapfsäule

Interaktionsfolge:

Schritt 1: Fahrer verlässt das Fahrzeug,

Schritt 2: Fahrer öffnet den Tankverschluss,

Schritt 3: Fahrer wählt die Zapfpistole aus, steckt sie in den Tankstutzen und betätigt den Starthebel, System liefert Treibstoff bis Tank voll oder Tanken abgebrochen wird und startet die Anzeige

Schritt 4: Fahrer entnimmt die Zapfpistole und hängt sie ein

Schritt 5: Fahrer verschließt den Tank.

Anwendungsfall: Tanken

Ausnahmen und Fehlerfälle (Extensions):

2: Wenn der Tankverschluss sich nicht öffnen lässt dann
Störungsbeseitigung

3: Wenn Pumpe und Zählwerk nicht starten dann
Mitteilung an Bedienpersonal

4: Wenn Treibstoff verschüttet wird dann Aufnahme und
Beseitigung des Treibstoffes

Enthaltene (Included) Anwendungsfälle:

ggf. Reinigung der Scheiben

Ausführungszeit: 1 bis 3 Minuten

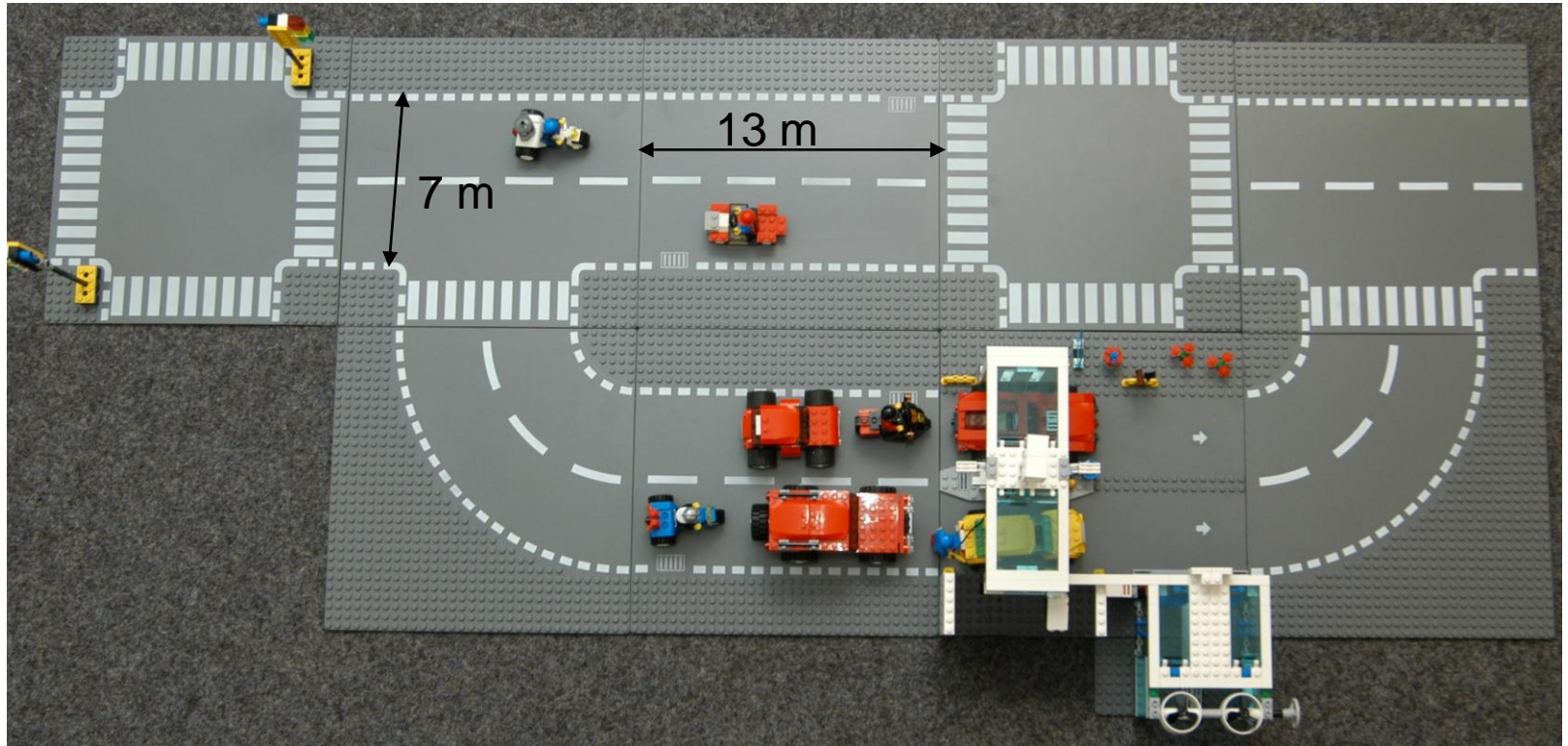
Übergeordnete Anwendungsfälle: Tankstellennutzung

Ereignisse: Tanken und Einkaufen

- Fahrzeug erreicht den Tankstellenbereich
 - Welche Zapfsäule bzw. Warteschlange wird ausgewählt?
- Fahrzeug erreicht Zapfsäule bzw. Warteschlange
- Fahrer steigt aus und beginnt mit dem Tanken
- Tanken beendet, zum Tankstellenshop gehen
 - Wird noch im Tankstellenshop eingekauft?
- Tankstellenshop erreicht/ Beginn Einkauf bzw. Kasse erreicht
- Beginn Bezahlen
- Ende Bezahlen
- Fahrzeug erreicht / Abfahrt / Freigabe der Zapfsäule
- Tankstellenbereich verlassen

Layout des Lego - Modells

Maßstab 1 : 50



Prozesszeiten: Tanken und Einkaufen

Prozess	Dauer (s)	Strecke (m)
Zwischenankunftszeit der Fahrzeuge	220 ± 100	
Fahrt von Eingang bis Zapfsäule	2,5 ± 0,5	18 ... 24
Tanken	120 ± 60	
Gehen von Zapfsäule zum Shop	15 ± 5	10 ... 13
Einkaufen (20%)	110 ± 100	
Bezahlen	60 ± 30	
Gehen von der Kasse zum Fahrzeug	20 ± 5	15 ... 19
Fahrt von Zapfsäule bis Ausgang	1,5 ± 0,5	5 ... 11

Rollenspiel: Tankstelle

Nr.	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11
1	142	144,6	144,6	227,4	239,7		244,7	244,7	261,5	281,5	283,1
2	188	189,7	283,1								
3	222										

E1: Erreichen Tankstelle ($ZAZ = 120 + 200 * RNI$)

E2: Erreichen Zapfsäule bzw. Warteschlange ($Zeit = 2 + RNI$)

E3: Beginn Tanken

E4: Ende Tanken ($Zeit = 60 + 120 * RNI$)

E5: Erreichen Shop/ Beginn Einkaufen ($Zeit = 10 + 10 * RNI$)

E6: Ende Einkaufen (E5 bzw. $Zeit = 10 + 200 * RNI$)

E7: Erreichen Kasse bzw. Warteschlange

E8: Beginn Bezahlen

E9: Ende Bezahlen ($Zeit = 30 + 60 * RNI$)

E10: Erreichen Fahrzeug ($Zeit = 15 + 10 * RNI$)

E11: Verlassen Tankstelle ($Zeit = 1 + RNI$)