

Anhang

Inhaltsverzeichnis

A	UML-Diagramme des Simulatorkerns.....	A-1
A.1	Klassendiagramme.....	A-1
A.2	Sequenzdiagramme	A-6
A.3	Statusdiagramme der Prozesse	A-11
B	Konfiguration der Streckentopologie	B-13
B.1	Satzformat der Testdatei	B-13
B.2	Konfiguration einer Querverschiebung	B-15

A UML-Diagramme des Simulorkerns

In diesem Anhang werden die UML-Diagramme angegeben, auf die zuvor im Text verwiesen wurde. Die Diagramme dokumentieren die verwendeten Klassen und ihre Interaktionen.

A.1 Klassendiagramme

Die Klassendiagramme verdeutlichen die Struktur der im Simulorkern eingesetzten Klassen und ihre Beziehungen untereinander.

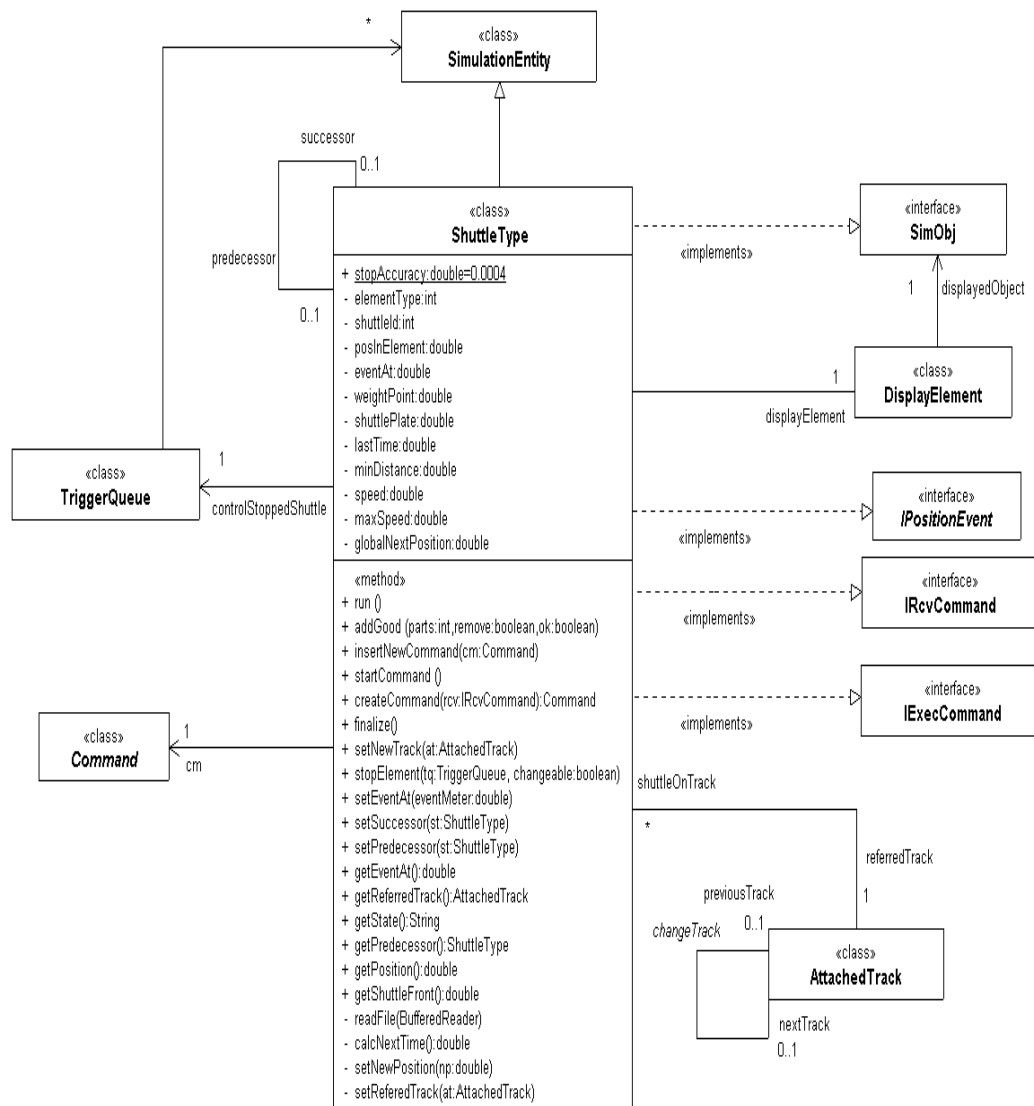


Bild 1: Das Klassendiagramm beschreibt die Modellierung eines Shuttles im Simulorkern.

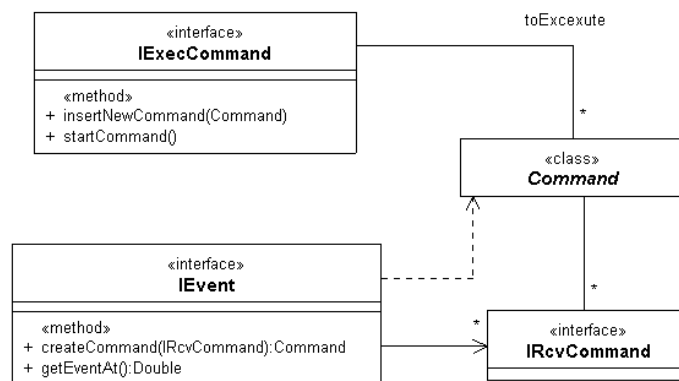


Bild 2: Das Klassendiagramm zur Befehlsausführung verdeutlicht den Einsatz des Befehlsmodells im Simulatorkern. Um die Entwicklung flexibel zu gestalten, wurden für den „Aufrufer“, „Empfänger“ und „Erzeuger“ jeweils allgemeine Schnittstellen eingesetzt.

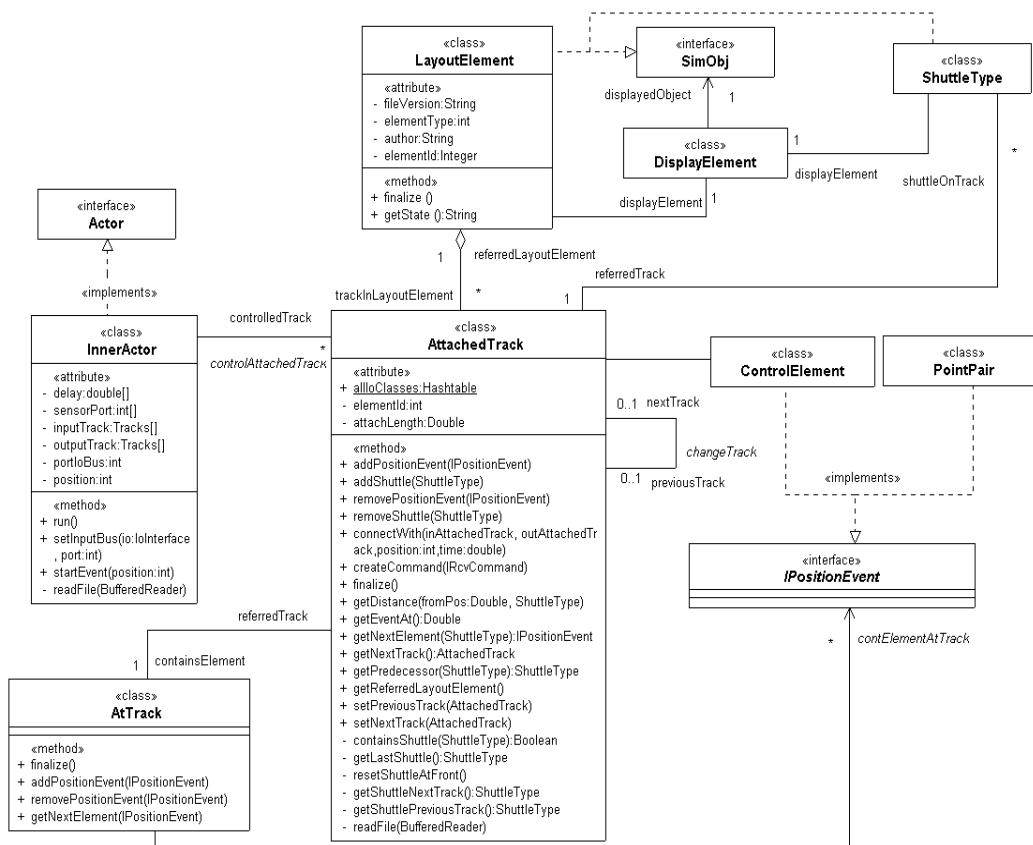


Bild 3: Das Klassendiagramm veranschaulicht die Modellierung der Streckenmodule unter Berücksichtigung einzelner Aspekte aus der Designphase.

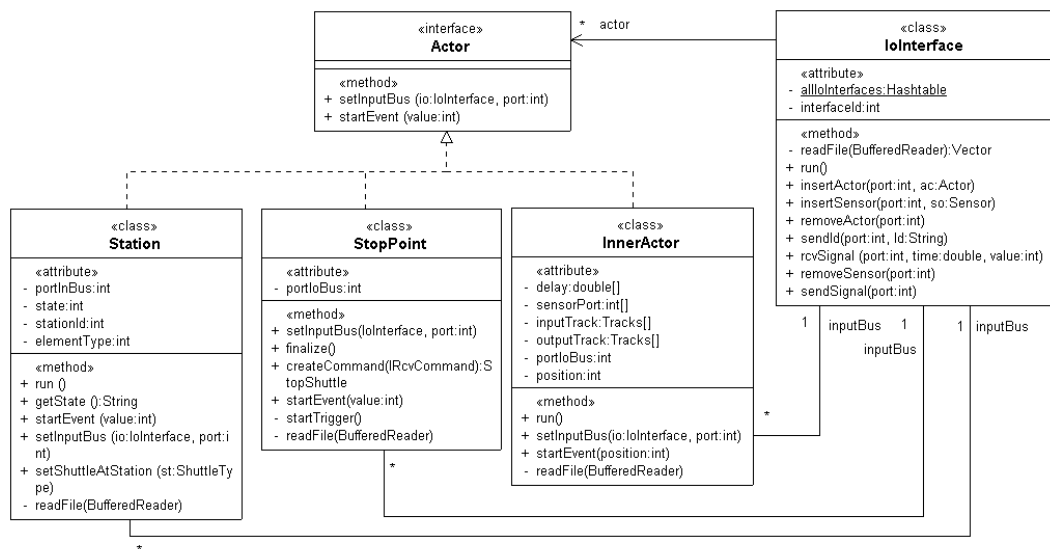


Bild 4: Das Klassendiagramm zeigt, dass die Modellelemente „Station“, „StopPoint“ und „InnerActor“ die Schnittstelle „Actor“ implementieren. Daher kann die Steuerung auf diese Elemente einwirken.

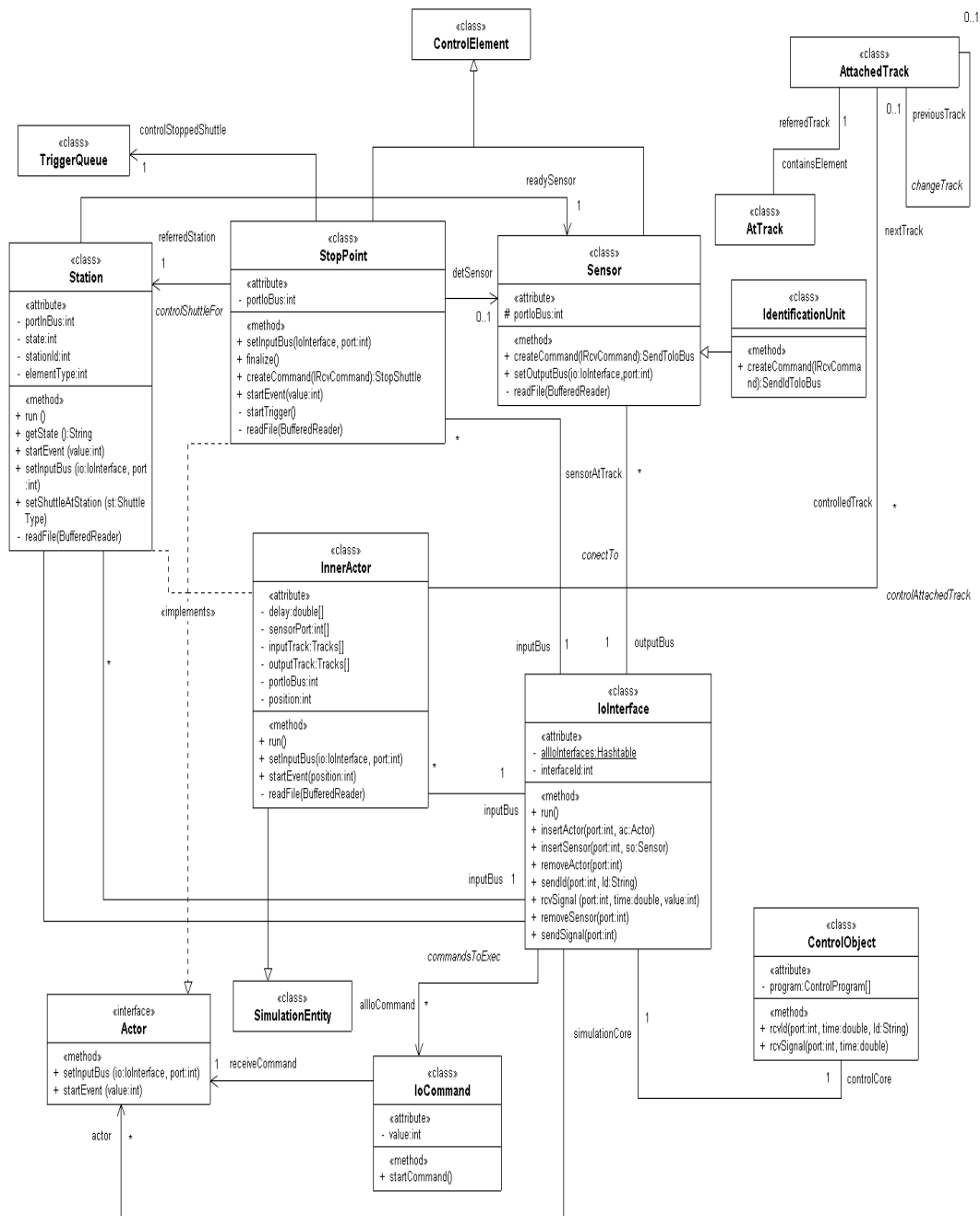


Bild 5: In diesem Klassendiagramm ist die Verbindung der Kontrollelemente mit den Prozessinterface angegeben. Die Schnittstelle zu den Steuerungsobjekten bildet die Klasse „ControlObject“.

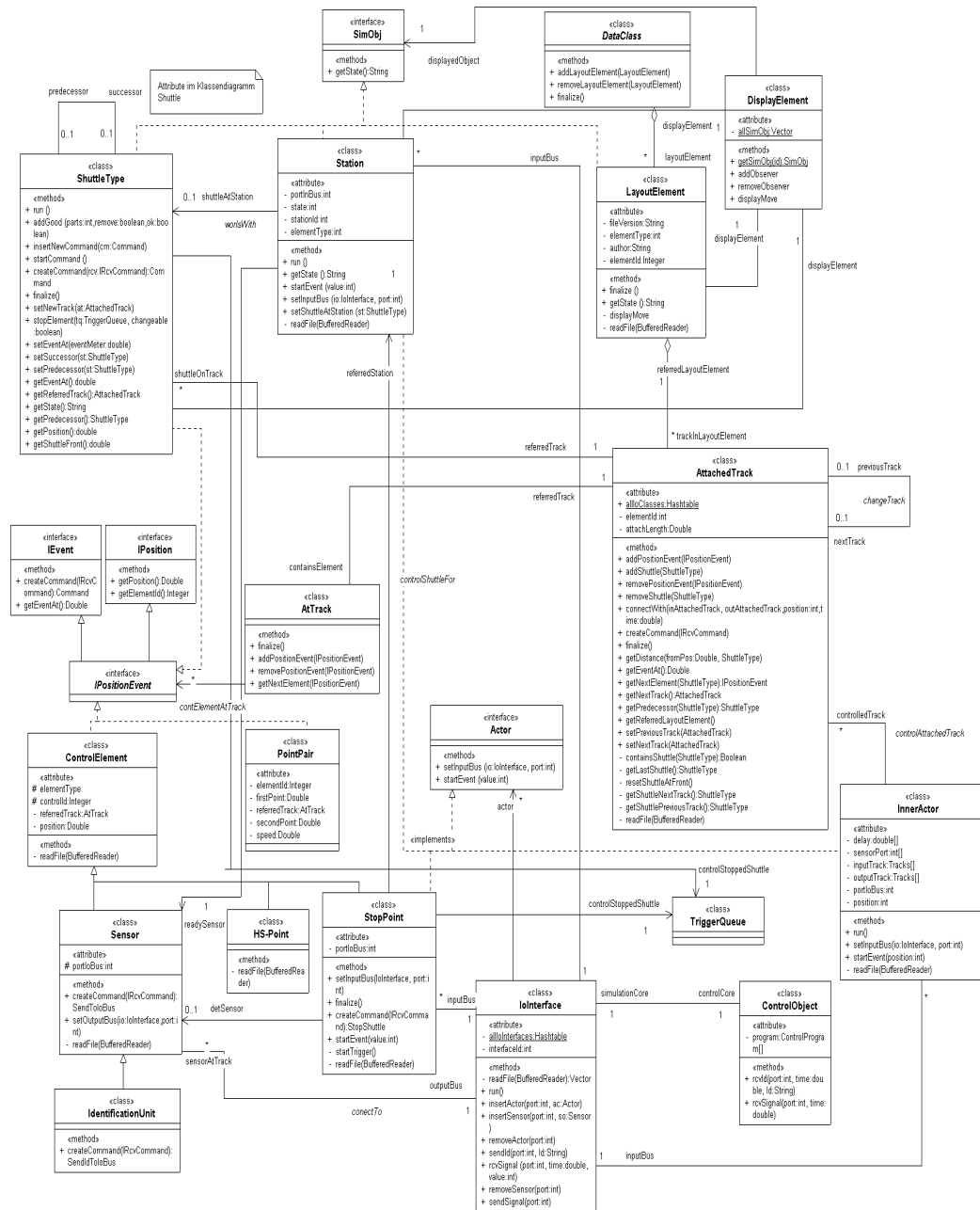


Bild 6: Dieses Klassendiagramm zeigt die im Simulatorkern modellierten Klassen und ihre Beziehungen.

A.2 Sequenzdiagramme

Da eine zeitliche Synchronisation der Ereignisse im Simulatorkern erforderlich ist, werden die Interaktionen der Objekte durch Sequenzdiagramme dargestellt.

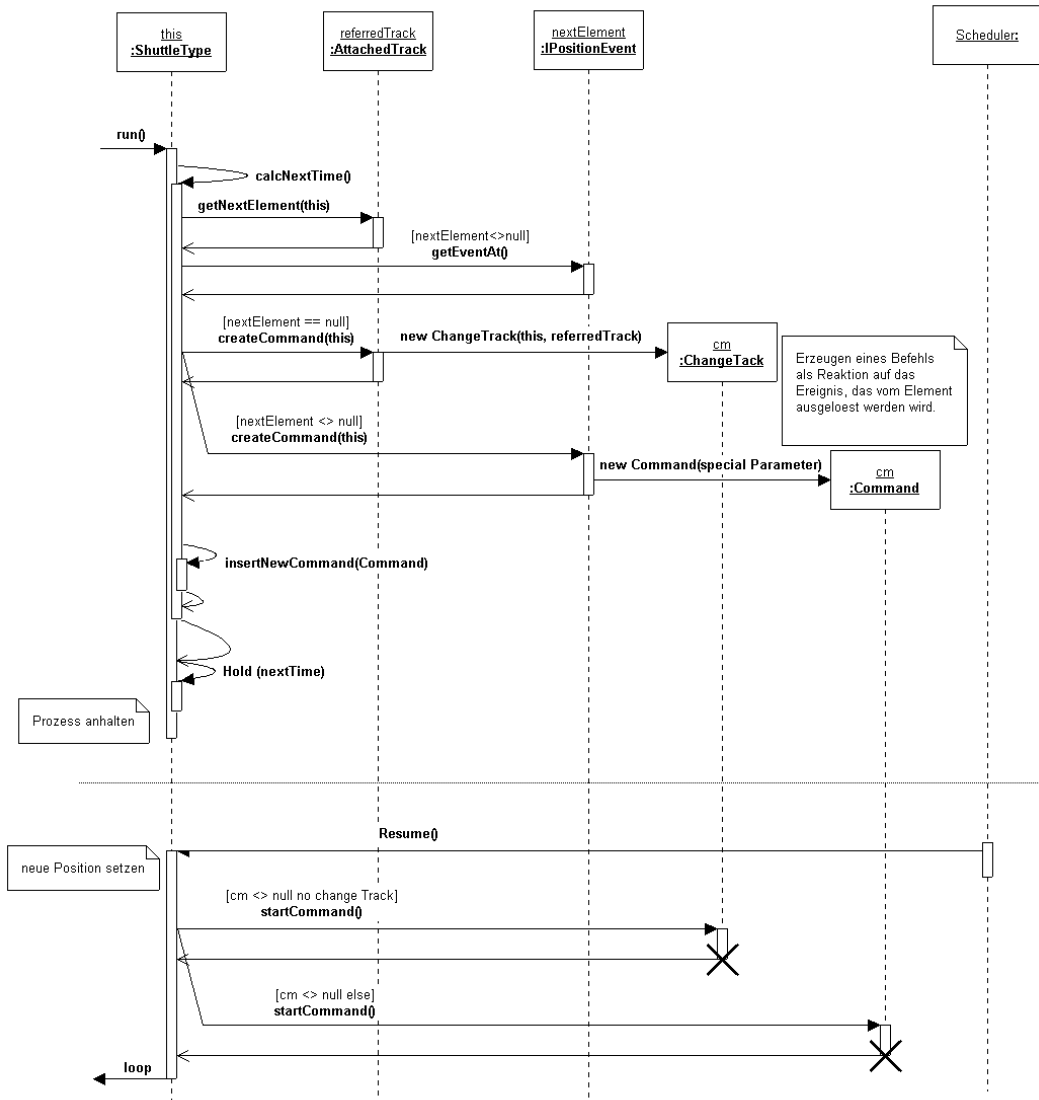


Bild 7: Das Sequenzdiagramm beschreibt den zeitlichen Verlauf des Shuttleprozesses.

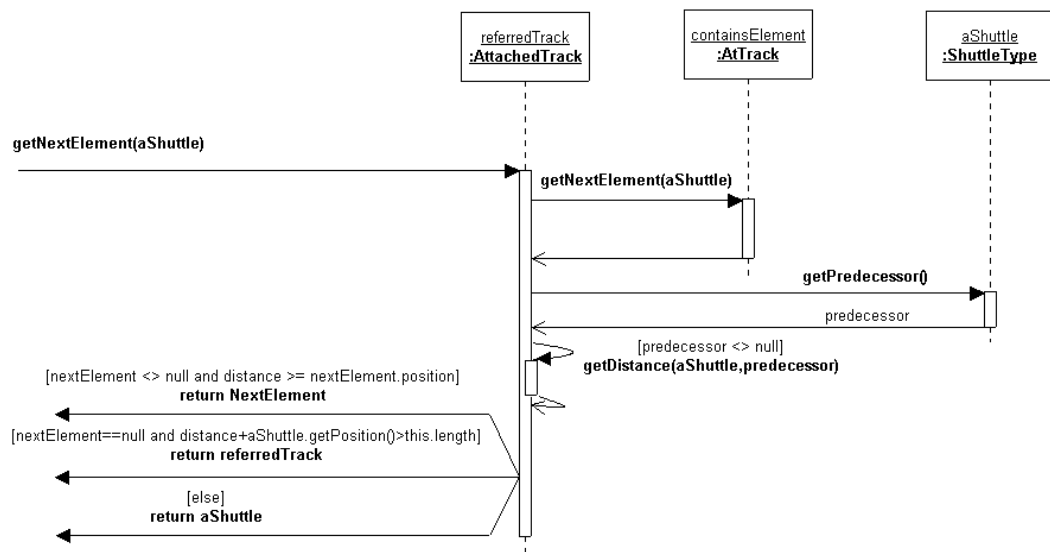


Bild 8: Diese Darstellung veranschaulicht den Ablauf zur Ermittlung des nächsten Elementes vor dem aktiven Shuttle in einem Sequenzdiagramm.

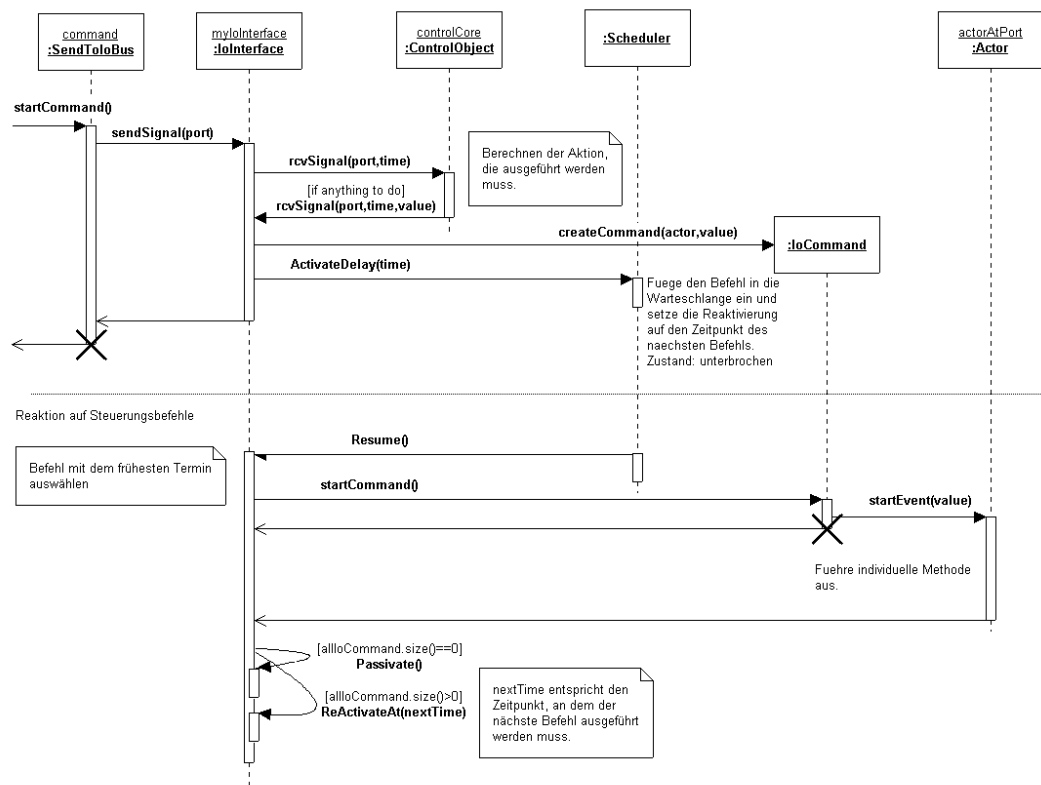


Bild 9: Das Sequenzdiagramm zeigt den Vorgang der Kommunikation zwischen dem Kern und den Steuerungsobjekten.

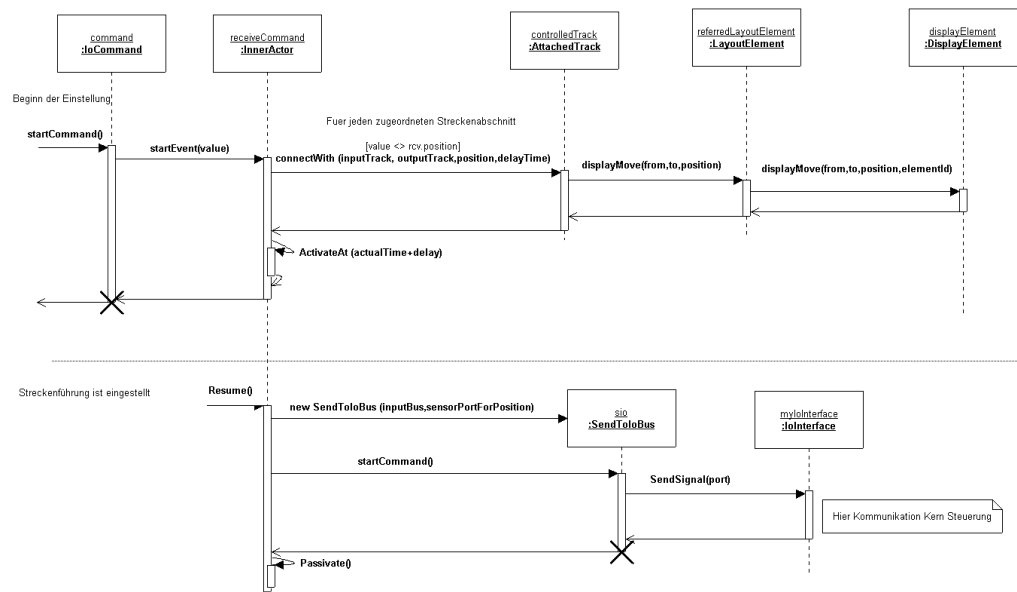


Bild 10: Das Sequenzdiagramm verdeutlicht die Veränderung der Streckenführung variabler Streckenteilabschnitte.

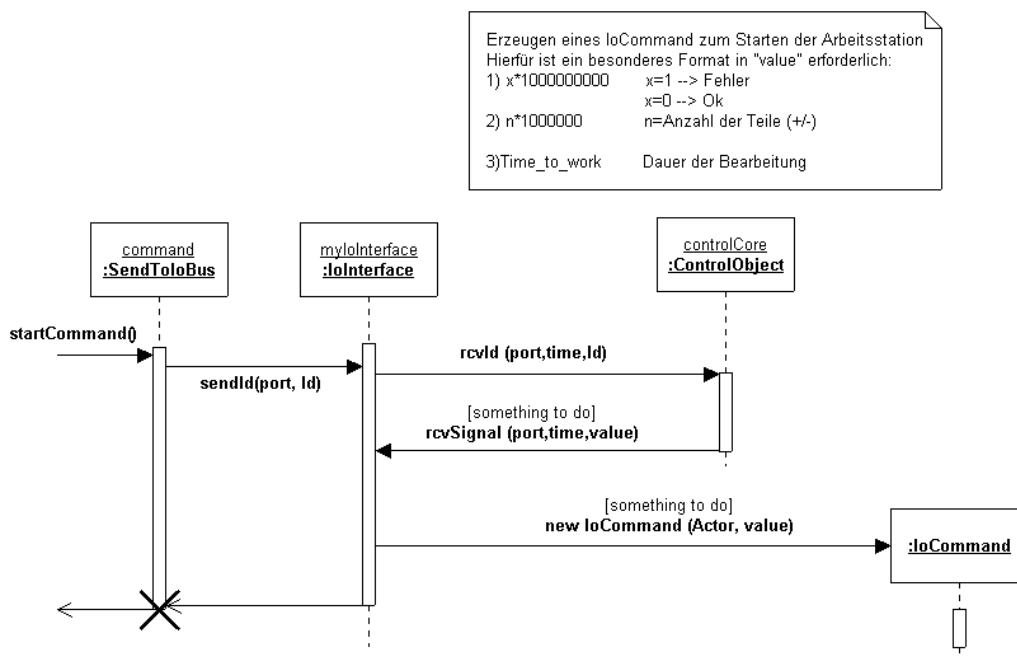


Bild 11: Dieses Sequenzdiagramm zeigt die Methodenaufrufe zum Beginn der Bearbeitung an einer Arbeitsstation auf.

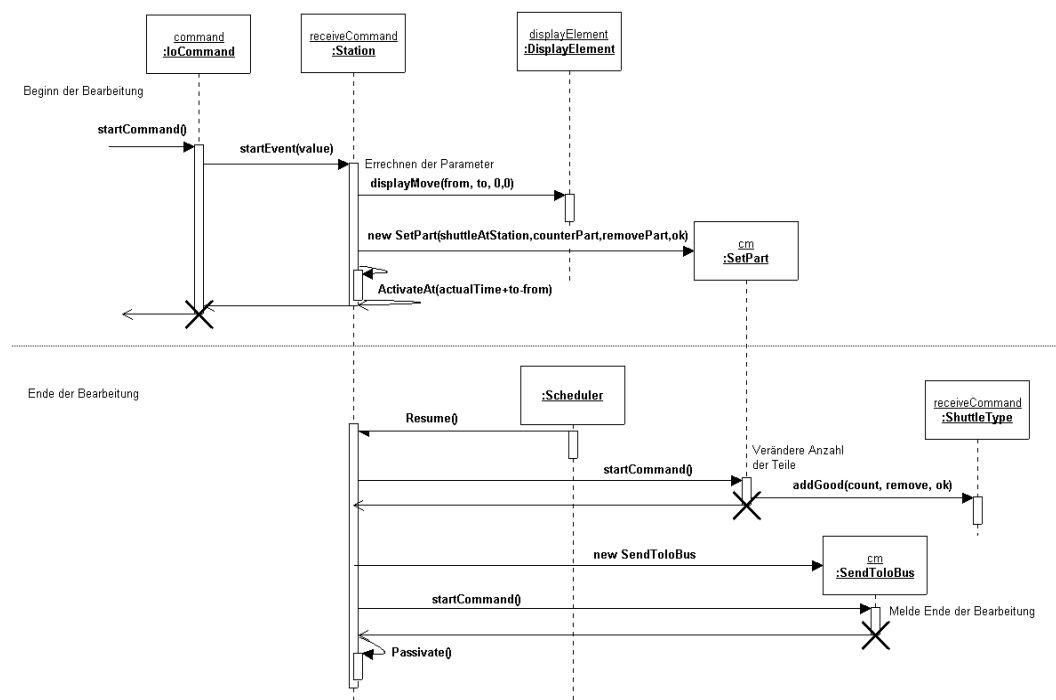


Bild 12: Das Sequenzdiagramm beschreibt den Verlauf der Bearbeitung an einer Station.

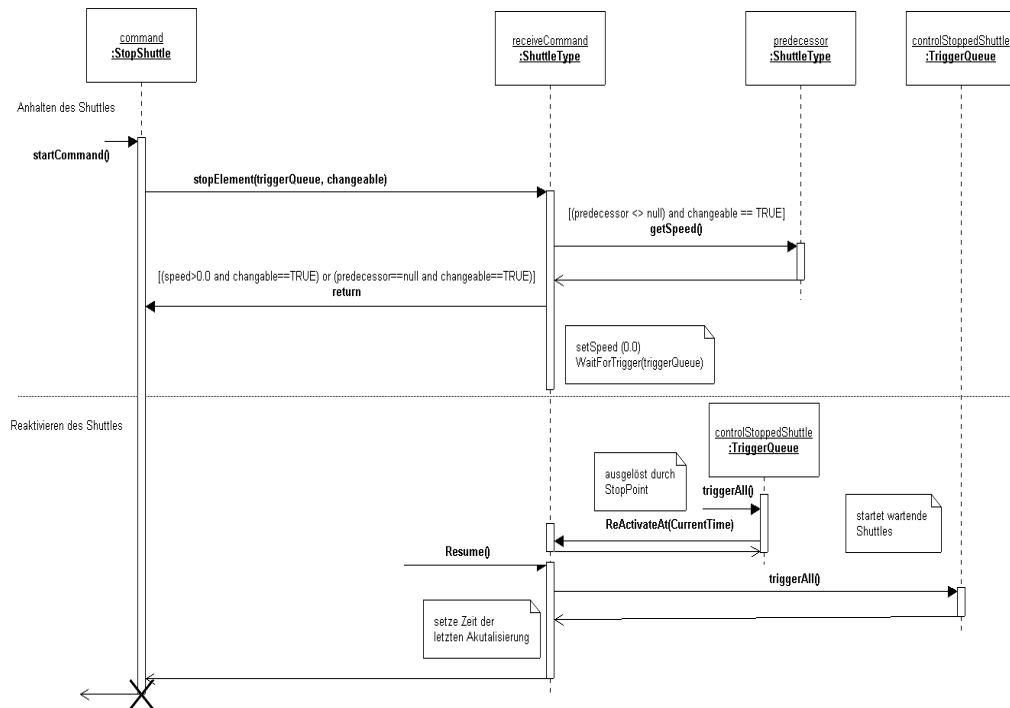


Bild 13: Das Sequenzdiagramm verdeutlicht die Ausführung eines Stoppbefehls.

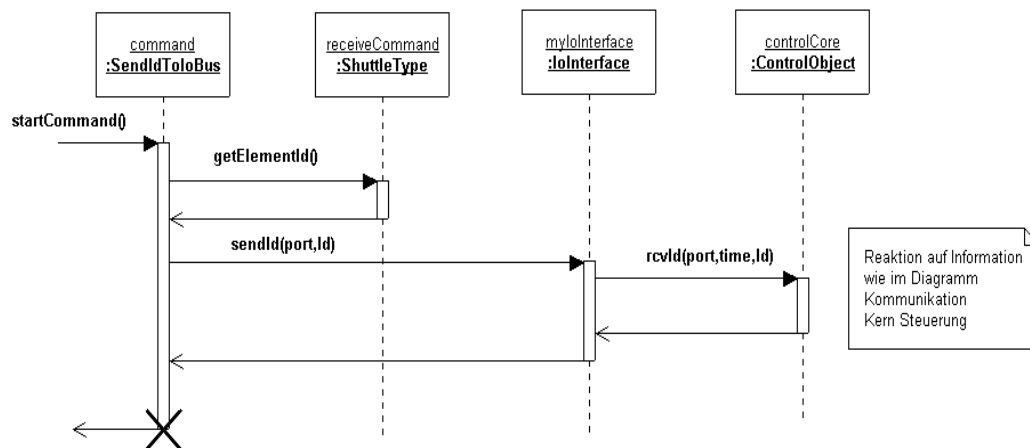


Bild 14: Aus diesem Sequenzdiagramm wird die Ausführung eines Befehls ersichtlich, der als Reaktion auf ein Ereignis zum Auslesen und Weiterleiten der Shuttleidentifizierung generiert wird.

A.3 Statusdiagramme der Prozesse

Die Statusdiagramme der Prozesse verdeutlichen den Zustand der Objekte in Bezug auf die Verwaltung durch die prozesskontrollierende Instanz.

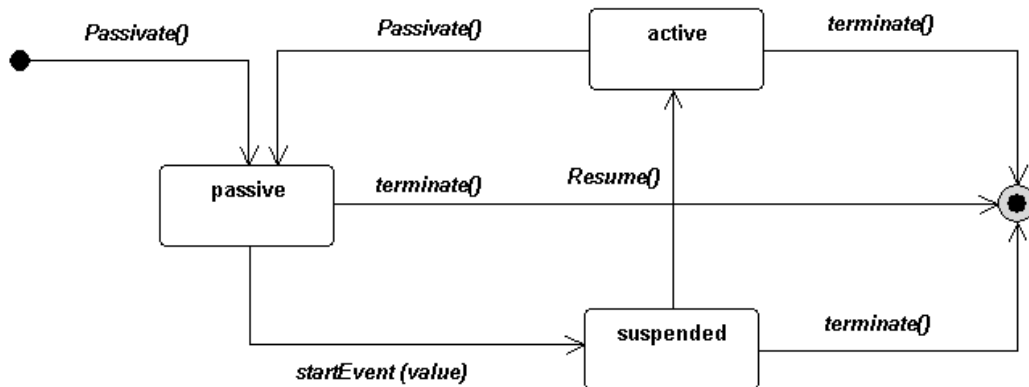


Bild 15: Dieses Statusdiagramm veranschaulicht den Zustand der Prozesse zur Modellierung der Antriebe.

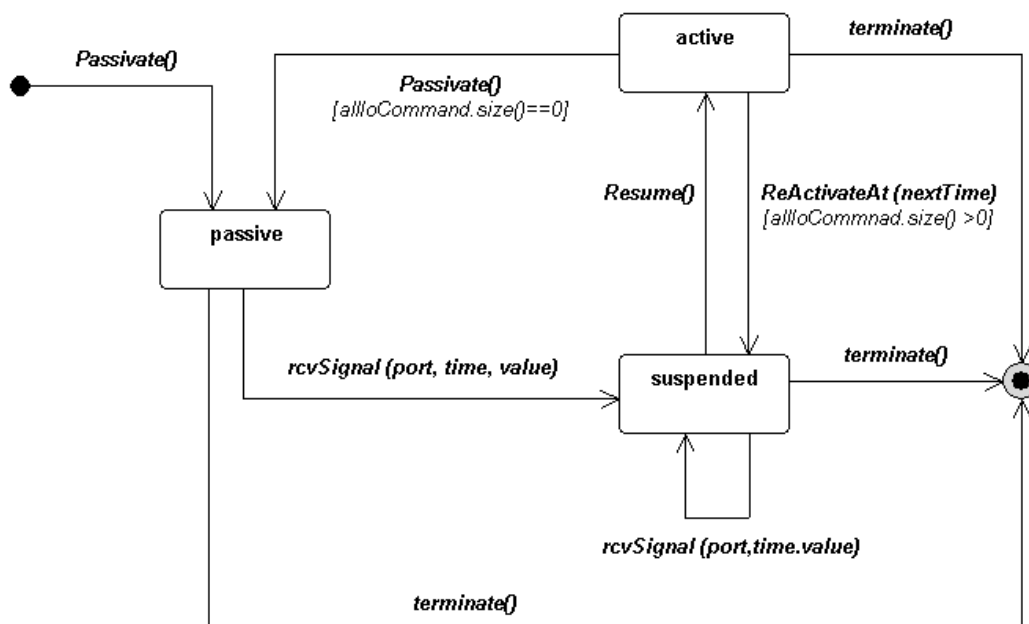


Bild 16: Das Statusdiagramm verdeutlicht die Zustände der Prozessschnittstelle.

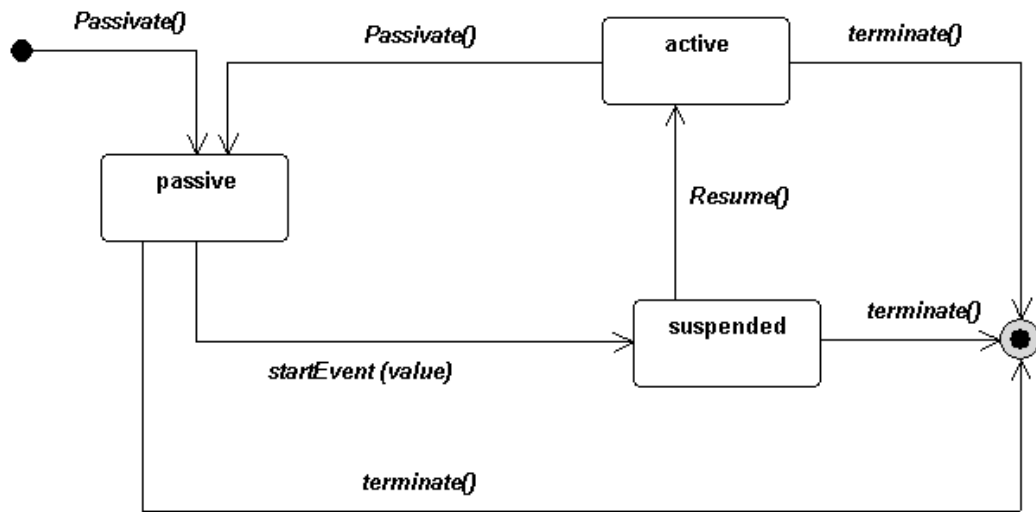


Bild 17: Statusdiagramm einer Arbeitsstation

B Konfiguration der Streckentopologie

Ausgehend vom allgemeinen Aufbau einer Datei zur Konfiguration eines Simulationslaufes, wird als Beispiel die Beschreibung des komplexen Streckenmoduls *Querverschiebung* angegeben.

B.1 Satzformat der Testdatei

Dieser Abschnitt beschreibt den Aufbau der Konfigurationsdatei eines Simulationslaufes. Im Vordergrund steht die Definition der Modellelemente des Simulatorenkerns.

Allgemeine Angaben:

- Alle Entfernungen werden in Metern angegeben
- Geschwindigkeiten in Meter/Sekunde
- Positionen beziehen sich auf den Beginn eines Streckenteilabschnittes

Element : an dieser Stelle kann das Modellelement *Element* konfiguriert werden
 (0..n) : Angabe kann beliebig oft angegeben werden
 (1) : genau eine Angabe
 (1..n) : mindestens eine Angabe erforderlich

Übergeordnetes Format:

IoInterface (0..n)

LayoutElement (1..n)

Steuerungsknoten IoInterface	Streckenmodul LayoutElement
begin IoInterface Nummer des Knotens begin program <u>Program</u> (0..n) end program end IoInterface	begin LayoutElement Dateiversion Elementtyp Elementidentifizierung Autor <u>attachedTrack</u> (1..n) <u>innerActor</u> (0 .. n)
Steuerprogramme Program	begin name3D Name end name3D
Applikationsname Portbelegung(1,2,3...)	end LayoutElement

Streckenteilabschnitte AttachedTrack	Antriebe variabler Streckenteile InnerActor
begin attachedTrack Länge Eigene Identifizierung Vorheriger Abschnitt Nachfolgender Abschnitt <u>shuttle (0..n)</u> <u>station (0..n)</u> <u>identificationUnit (0..n)</u> <u>sensor (0..n)</u> <u>stopPoint (0..n)</u> end attachedTrack	begin innerActor PortNummer Zugehöriger Steuerknoten Anzahl zugeordneter Streckenteile Aktuelle Position Wiederhole Zugehöriger Track Wiederhole Von track Nach track bis endPosition bis end controlledTracks Wiederhole Zeit Verzögerung Sensorsignal an Port Bis end delaySensor end innerActor

Shuttle	Station
begin shuttle Identifizierung Shuttletyp Position Geschwindigkeit Länge der Shuttleplatte Punkt zur Repräsentation (von Beginn) Höchstgeschwindigkeit Minimaler Abstand zwischen Shuttles Name für Beobachter end shuttle	Begin station Status Identifizierung Stationstyp Port für Fertigmeldung Nummer des Steuerknotens <u>sensor //melde Bearbeitungsende</u> {begin name3D Name der Station} end station

Sensoren Sensor	Identifizierungseinheit identificationUnit	Start-Stopeinheit StopPoint
begin sensor Identifizierung Position Typ des Sensors end controlElement Port Steuerknotennummer end sensor	begin identification Identifizierung Position Transpondertyp end controlElement Port Steuerknotennr. end sensor	begin stopPoint Identifizierung Position Typ end controlElement Port Steuerknotennr. <u>Station (0..1)</u> <u>Sensor (0..1)</u> end stopPoint

B.2 Konfiguration einer Querverschiebung

*Beschreibung eines Testlayout

*Testlayout fuer eine Querverschiebung

begin IoInterface

1

begin program

QuerV2

100,101,102,103,104,105,106,110,111,112,113

end program

end IoInterface

****Quer1	end shuttle	begin attachedTrack	112
begin LayoutElement	begin identification	3	1
test	1502	101006	begin sensor
20	1.9	0	1502
10501	1	0	0.5499999
Ralf	end controlElement		1
begin attachedTrack	101	begin sensor	end controlElement
2	1	1502	102
101001	end sensor	1.9	1
101004		1	end sensor
0	begin stopPoint	end controlElement	end stopPoint
begin identification	502	104	end attachedTrack
1502	1.95	1	
1.9	1	end sensor	begin innerActor
1	end controlElement		113
end controlElement	111	begin stopPoint	1
100	1	502	2
1	end stopPoint	2.95	1
end sensor	end attachedTrack	1	101003
		end controlElement	101001
begin stopPoint	begin attachedTrack	1001	101005
502	3	1	0
1.95	101005	end stopPoint	0
1	101004	end attachedTrack	end Positions
end controlElement	0		101004
110		begin attachedTrack	101002
1	begin sensor	1	101006
end stopPoint	1502	101003	101001
end attachedTrack	1.9	0	101005
	1	0	end Positions
begin attachedTrack	end controlElement	end attachedTrack	end controlledTracks
2	103		2
101002	1	begin attachedTrack	106
0	end sensor	1	2
0		101004	105
begin shuttle	begin stopPoint	101005	end delaySensor
100001	502	101001	end innerActor
101	2.95		begin name3D
0.2	1	begin stopPoint	Quer
0.25	end controlElement	502	end name3D
0.2	1000	0.55	end LayoutElement
0.15	1	1	****Ende Quer1
0.25	end stopPoint	end controlElement	
0.130	end attachedTrack		
Shuttle1			

