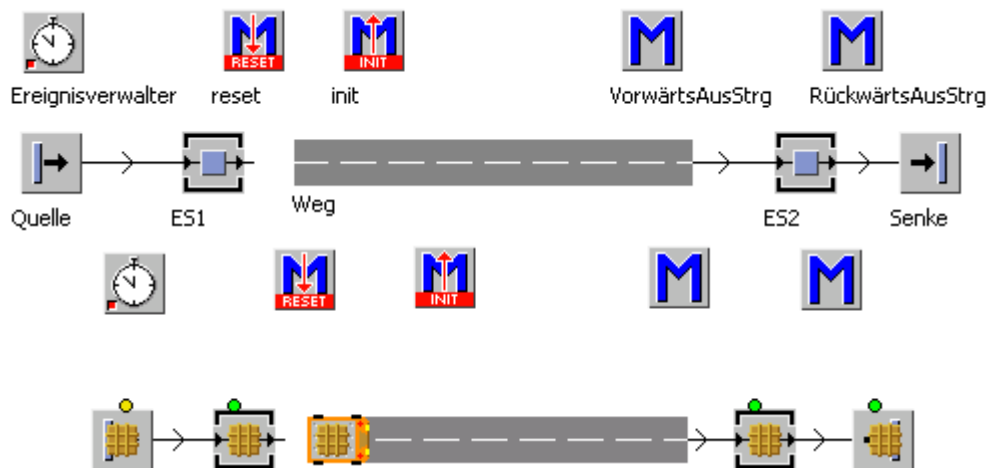


Ein Transportsystem mit passiven Elementen modellieren



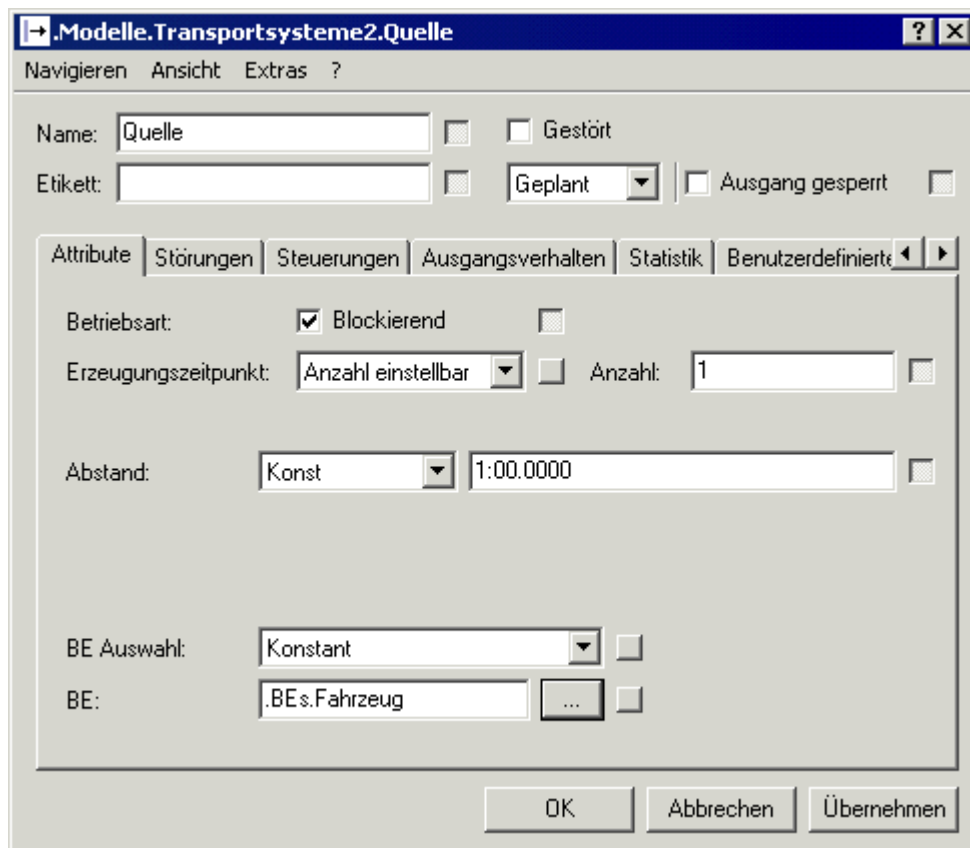
Wenn Sie Transportsysteme mit passiven Elementen modellieren, setzen Sie den *Weg* und das *Fahrzeug* ein.

Da der *Weg* ein passives Objekt ist, können wir ihn nicht einfach mit zwei Stationen verbinden. Wir müssen vielmehr sicherstellen, daß ein *Fahrzeug* zur Verfügung steht. Außerdem müssen wir dieses *Fahrzeug* beladen und entladen. Um zu verhindern, daß BEs automatisch von der vorangehenden Station auf den *Weg* umlagern, in unserem Beispiel von *ES1*, haben wir sie nicht mit einer *Kante* verbunden.

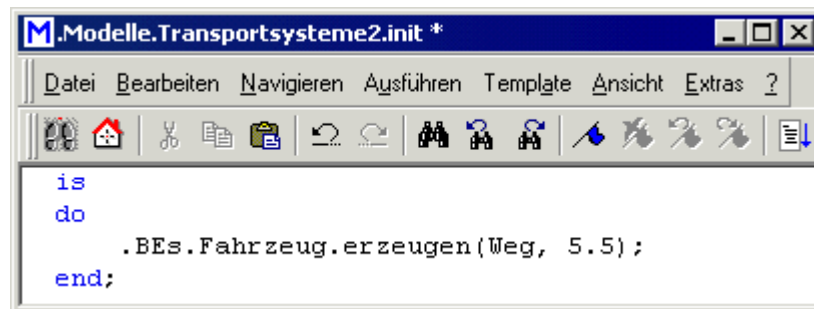


Um ein *Fahrzeug* zu erzeugen und einzusetzen, können Sie eine der folgenden Aktionen ausführen:

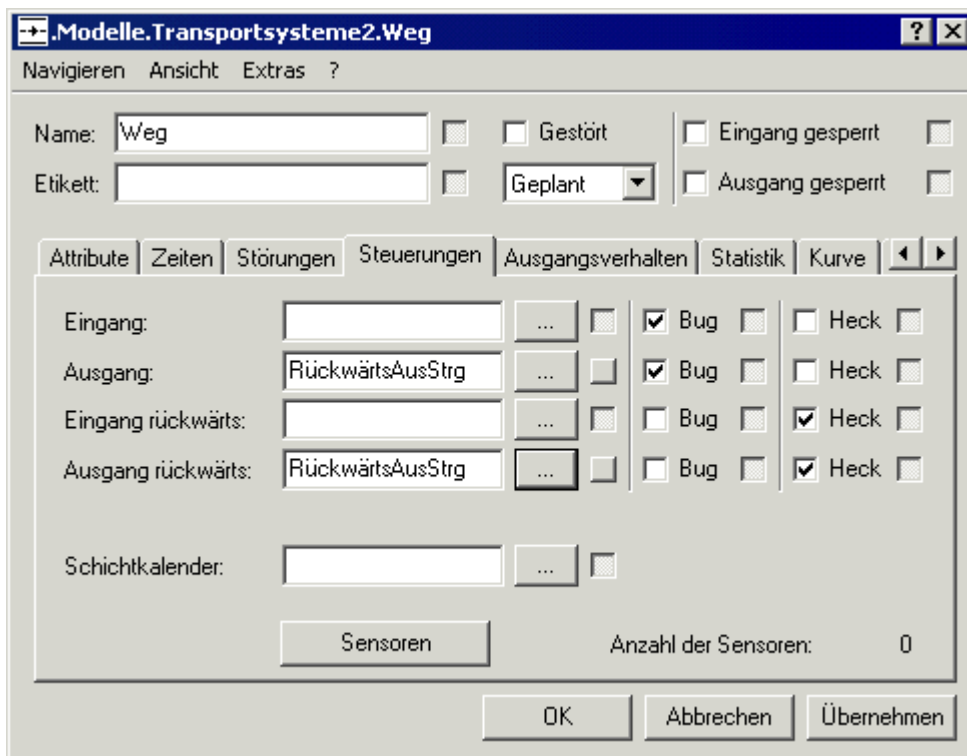
- Die *Quelle* benutzen: Wählen Sie **Attribute > Erzeugungszeitpunkt > Anzahl einstellbar** und tragen Sie eine **Anzahl** von 1 ein. Damit sie Fahrzeuge erzeugt, wählen Sie **BE > .BEs.Transporter**.



- Die SimTalk-Methode *erzeugen* benutzen und diesen Befehl in die *init*-Methode Ihres Simulationsmodells eintragen: `.BEs.Fahrzeug.erzeugen(Weg);`
Dies setzt ein *Fahrzeug* am Ende des *Wegs* ein. Wenn Sie ein *Fahrzeug* an einer anderen Stelle auf dem *Weg* einsetzen möchten, tragen Sie eine Position in die Methode ein. Dies Beispiel erzeugt ein *Fahrzeug* an der Position 5,5 Meter des *Wegs*.



Da das *Fahrzeug* Teile von Station zu Station transportieren soll, *ES1* und *ES2* in unserem Beispiel, müssen wir diese Teile an der Station *ES1* auf das *Fahrzeug* laden und es an der Station *ES2* wieder davon abladen. Dafür programmieren wir eine Steuerung in einer Methode und tragen deren Namen in das entsprechende Textfeld auf der Registerkarte **Steuerungen** ein.

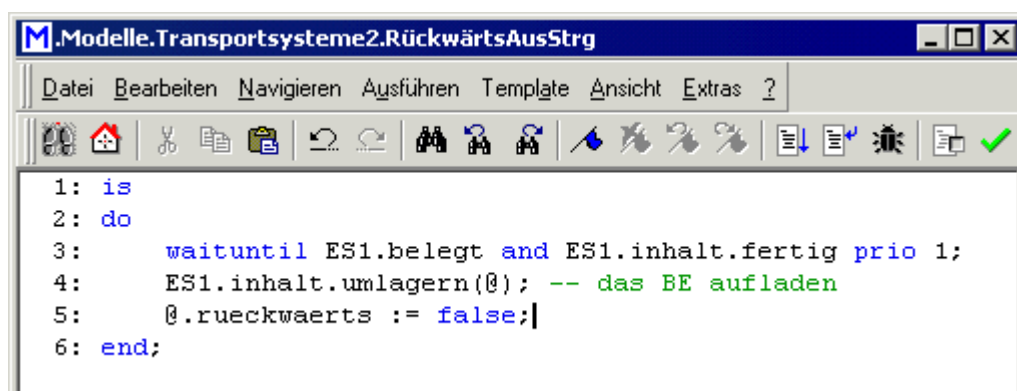


Wenn Sie keine Steuerung eintragen, bewegt sich das *Fahrzeug* nur bis zum Ende des *Wegs* und hält dort an. Damit das *Fahrzeug* an den Anfang des *Wegs* zurückfährt, wenn es die Teile abgeladen hat, aktivieren Sie das Kontrollkästchen **Bug** und tragen Sie diese Anweisungen in die **Ausgang**-Steuerung ein, *RückwärtsAusStrg* in unserem Beispiel:

```
if @.leer then
  @.rueckwaerts := true;
end;
```

Sobald das *Fahrzeug* rückwärts an den Anfang des *Wegs* gefahren ist, aktiviert eM-Plant die heckgesteuerte Ausgangssteuerung, wenn Sie das Kontrollkästchen **Heck** aktivieren und den Namen der Methode in das Textfeld **Ausgang rückwärts** eintragen. Diese Steuerung muß diese Aufgaben erfüllen:

- Warten, bis ein BE bereit ist aus der Station, *ES1* in unserem Beispiel, auszutreten.
 - Die BEs auf das *Fahrzeug* aufladen.
 - Das *Fahrzeug* ans Ende des **Wegs** fahren lassen.
- In unser Beispiel haben wir diese Anweisungen eingetragen.



Zeile 3 teilt eM-Plant mit zu warten, bis sich ein BE auf der Station *ESI* aufhält und ganz bearbeitet ist. Zeile 4 lädt das BE auf das *Fahrzeug*. Beachten Sie, daß wir den anonymen Bezeichner *@* benutzen, um das *Fahrzeug* anzusprechen. Zeile 5 veranlaßt das *Fahrzeug* wieder vorwärts zum Ende des *Wegs* zu fahren.

Wenn das *Fahrzeug* das Ende des *Wegs* erreicht hat, müssen Sie diese Aufgaben lösen:

- Warten bis die Station, *ESI* in unserem Beispiel, leer ist.
- Das *Fahrzeug* entladen.
- Das *Fahrzeug* an den Anfang des *Wegs* fahren lassen.

Da das *Fahrzeug* anfangs ans Ende des *Wegs* fährt und dann die folgende Ausgangssteuerung aktiviert, während es entladen wird, fragen wir ab, ob es beladen oder leer ist.

In unser Beispiel haben wir diese Anweisungen eingetragen.

```
1: is
2: do
3:   if @.leer then
4:     @.rueckwaerts := true;
5:   else
6:     waituntil ES2.leer prio 1;
7:     @.inhalt.umlagern(ES2);
8:     @.rueckwaerts := true;
9:   end;
10: end;
```

Gehe zu [Einen Querverschiebewagen modellieren](#)

Zurück zu [Transportsysteme modellieren](#)

