


PR1	Alternative zur Mehrfachvererbung	gds2 
-----	-----------------------------------	--

Java Special:

---

### Problem:



Unser Getränkelastwagen ist in unserem Fuhrpark von der Klasse Fahrzeuge abgeleitet worden.

Nun soll wie in einer anderen Applikation (die machen irgendwas mit Schiffen, Booten ???) auch die physikalischen Fähigkeiten der Verzögerung und Beschleunigung realisiert werden.

Die Vererbung der möglichen Methoden *verzoegern()* und *beschleunigen()* funktioniert nicht, da der Getränkelastwagen bereits von Kraftwagen erbt.


Wie können wir nun *verzoegern()* und *beschleunigen()* trotzdem „vererben“ ?



### Information:

- In Java ist nur Einfachvererbung möglich, soll eine Klasse auf einer Ebene von mehreren Typen erben, geht das durch die Einfachvererbung nicht. Da es aber möglich sein soll, dass in der objektorientierten Modellierung eine Klasse mehrere Typen in einem Schritt besitzt, gibt es das Konzept der Schnittstelle (engl. **interface**). Eine Klasse kann dann neben der Oberklasse eine beliebige Anzahl Schnittstellen implementieren und auf diese Weise weitere Typen sammeln.
- **Interface deklarieren:**

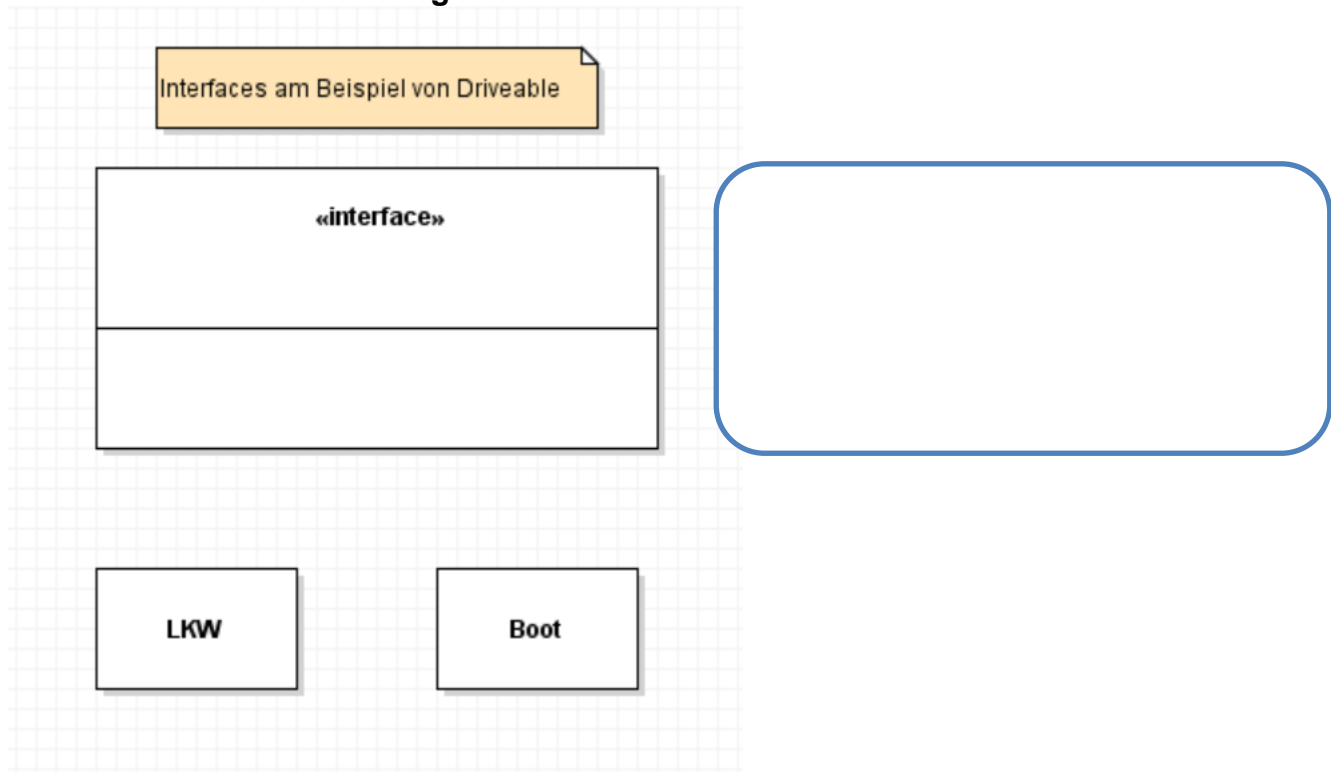
- **Interface benutzen / implementieren**

PR1	Alternative zur Mehrfachvererbung	gds2 
-----	-----------------------------------	--

- Eine Klasse erweitert genau eine Superklasse, aber sie kann mehrere Schnittstellen implementieren.
- **Hinweis/ Eclipse Spezialität:** Der Name einer Schnittstelle endet oft auf -ble (Accessible, Adjustable, Runnable).

## Planung:

- Interfaces in UML Diagrammen verwenden



## Durchführung:

- Implementierung des obigen Beispiels in Eclipse

## Kontrolle:

- Implementierung einer StarterKlasse in Eclipse um Boote und LKWs zu verzögern und beschleunigen