



## Entwurfsentscheidungen

Wie in der Analyse nutze ich in meinem Entwurf die Fabrik-Methode zur Erstellung der Körperteile. Jedoch habe ich mich zur besseren Übersicht dazu entschieden, statt einzelner Fabrikmethoden für jedes Körperteil und jedes Gelenk eine eigene Fabrik-Unterklasse zu schreiben. Dadurch kann ich gemeinsame Schritte bei der Erzeugung, beispielsweise eines Körperteils, in der Methode *erzeugeKoerperteil* zusammenfassen und konkrete Anweisungen, z.B. hinsichtlich Länge und Radius des Körperteils, in der jeweiligen Implementierung der Methode *erzeugeKoerper* angeben.

Da ich auf die Implementierung der Klassen *Physik*, *ODEPhysik*, *Grafik* und *DrawstuffGrafik* verzichtet habe, habe ich für Körperteile, Gelenke, die Welt und die Simulation jeweils eine Oberklasse implementiert, die jeweils eine Schnittstelle für die einzelnen Objekte angibt, und eine konkrete Unterklasse geschrieben, die ODE und Drawstuff implementiert.

Wie bereits erwähnt, kann ich größtenteils auf anonyme Zeiger und deren Cast verzichten. Jedoch entsteht ein etwas höherer Aufwand, wenn die Bibliotheken ausgetauscht werden, da nun vier statt zwei neue Unterklassen implementiert werden müssen.

Zudem besitzen die einzelnen Fabriken nun die jeweilige Unterklasse (*ODEKoerperteil*, *ODEGelenk*) als Rückgabotyp. Der Aufwand, diese beim Austausch der Bibliotheken zu ändern, ist aber vernachlässigbar.

Es wäre sinnvoll, für den Roboter und die Welt mit dem Singleton-Pattern zu implementieren, da zu beiden Klassen zur Laufzeit jeweils nur eine Instanz existieren darf. Da ein Singleton jedoch ein Anti-Pattern ist, das eine Reihe von Nachteilen, wie die erschwerte Test- und Wartbarkeit des Codes, mit sich bringt, habe ich in meiner Implementierung darauf verzichtet.