



Probleme und Lösungsansätze

Problem 1

Die Aufgabenstellung verlangte von uns, dass es zwei Arten von Produkten gibt. Zum einen die Großhändlerprodukte, welche die Angebote der Zwischenhändler des Bioladens darstellen. Zum anderen unsere eigenen Produkte, die wir selber anbieten und verwalten können müssen. Dazu kommt, dass nicht alle Eigenschaften zwischen eigentlich gleichen Produkten gleich sind, wie zum Beispiel Einkaufspreis oder Mindestbestellmenge. So sahen wir uns vor das Problem gestellt, wie wir unseren Klassen aufbauen und fanden zwei gleichermaßen mit Vor- und Nachteilen verbundenen Lösungsansätze.

Die erste Variante wäre gewesen, alle Vorgaben in eine Klasse zu stecken. Man würde alle gleichbleibende Werte, wie zum Beispiel Basiseinheit oder die Produktkategorie, als eigene Werte speichern, und alle vom Großhändler abhängigen Werte zusammen mit diesem in eine Map speichern. Zusätzlich müsste man dann allerdings eventuell noch eine Art Flag mitführen, welche anzeigt, ob eines dieser Produkte von uns angeboten wird.

Die andere Variante trennt die beiden Konzepte Großhändlerprodukt und von-uns-angebotenes-Produkt vollständig bis auf eine Liste in unseren Produkten, welche die entsprechenden Großhändlerprodukte referenziert.

[Durchstreichen Animation]

Wir haben uns letzten Endes für die zweite Variante entschieden, da sie uns als die einfacher zu handelnde erschien.

[Nächste Folie (die mit dem Diagramm)]

Hier im Diagramm sehen wir nun die Umsetzung dieser Lösung. Wir haben zwei getrennte Klassen für die beiden Konzepte. Die einzige Verbindung ist ****diese**** die DistributorProducts referenzierende Liste *[Anm.: Ganz unten in den Attributen]*. Einige Attribute doppeln sich zwar so, jedoch ist durch die Trennung unter anderem auch eine einfachere Weiterverwendung in anderen packages, wie zum Beispiel der Bestellung, gegeben.

Mit freundlichen Grüßen

Adrian Kulisch

PS: Das Diagramm ragt in die Fußzeile rein.