**Eingabe der Daten in für die Erstellung des Graphen**

Rahmenbedingungen:  
Es ist herauszufinden, wie die Daten aus der Use-Case Beschreibungsdatei am besten an die Schnittstelle für die interne Use-Case Repräsentation übergeben werden soll.

Entwurfsalternativen:

1. Übergabe in einem Dictionary  
   (mehrere Alternative Flows können schlecht zugegriffen werden (gleicher Name in einem Dictionary nicht erlaubt))
2. Übergabe als Liste von Tupeln  
   (Datentypen von Tupeln können variieren (String, Tupel, etc.)
3. Übergabe der "einfachen" String Paramter über Properties und der Flows über Methoden

Ergebnis:  
Es wird die Alternative 3 verwendet. Begründung siehe oben.

**Weitergabe der Graph-Daten**

Rahmenbedingungen:  
Es ist zu klären, wie andere Komponenten der Anwendung Daten von der UseCase-Klasse erhalten sollen, nachdem der Graph erstellt wurde.

Entwurfsalternativen:

1. Jede Klasse, die Daten braucht erhält eine Referenz auf den UseCase vom Controller
2. Der UseCase verbreitet seine Daten über Eventsan alle, die die Daten brauchen  
   (jeder muss dafür eine Referenz auf den UseCase haben, also geht auch die andere alternative)
3. Der UseCase verbreitet seine Daten über ein Messaging-System an alle, die die Daten brauchen  
   (es existiert noch keine Notwendigkeit für ein globales Messaging-System)
4. Der UseCase kennt alle, die Daten brauchen und übergibt diese direkt  
   (zu starke Kopplung (Klassenbindung))

Ergebnis:  
Es wird die Alternative 1 verwendet. Begründung siehe oben.

**Flow Repräsentation**

Rahmenbedingungen:  
Es ist zu klären, ob es eine Klasse Flow für jeden Typ von Flow gibt oder ob es eine gemeinsame Basisklasse gibt, von der jeder Flow Typ abgeleitet ist.

Entwurfsalternativen:

1. Jeder Flow Typ hat die gleiche Klasse (Flow), bei der teilweise Vorhersagen über die Struktur abhängig vom Typ getroffen werden können (Basic Flow hat zum Beispiel keine Reference Flows)
2. Jeder Flow Typ hat seine eigenen Klasse, die von Flow abgeleitet ist.

Ergebnis:  
Es wird die Alternative 1 verwendet. Die Entscheidung wurde getroffen, da der Typ nur die Belegung einzelner Eigenschaften variiert, die für alle Flow Typen allerdings haben können. Weiter reduziert diese Alternative den Entwicklungsaufwand.