

Pattern/Prinzip	Beschreibung
<i>Information Expert</i>	Ein allgemeines Prinzip des Objektentwurfs und der Zuweisung von Verantwortlichkeiten; weisen Sie Verantwortlichkeiten dem <i>Information Expert</i> zu – der Klasse, die über die zu deren Erfüllung erforderlichen Informationen verfügt.
<i>Creator</i>	Wer erstellt? (Anmerkung: <i>Factory</i> ist eine gebräuchliche alternative Lösung.). Übertragen Sie Klasse B die Verantwortung, eine Instanz von Klasse A zu erstellen, wenn eine der folgenden Aussagen wahr ist: 1. B enthält A. – 2. B aggregiert A. – 3. B enthält Initialisierungsdaten für A. – 4. B zeichnet A auf. – 5. B ist mit A bei der Verwendung eng verbunden.
<i>Controller</i>	Welches erste Objekt jenseits der UI-Schicht empfängt und koordiniert (»kontrolliert«) eine Systemoperation? Weisen Sie die Verantwortlichkeit einem Objekt zu, auf das eine der folgenden Aussagen zutrifft: Repräsentiert das Gesamtsystem; ist ein »Root«-Objekt; ist ein Gerät, in dem die Software läuft; ist ein größeres Subsystem; (dies sind alles Varianten von <i>Facade Controller</i> ). Repräsentiert ein Use-Case-Szenario, in dem die Systemoperation abläuft (ein <i>Use Case</i> oder <i>Session Controller</i> ).
<i>Low Coupling</i> (evaluativ)	Wie kann der Einfluss von Änderungen verringert werden? Weisen Sie Verantwortlichkeiten so zu, dass die (nicht erforderliche) Kopplung gering bleibt. Verwenden Sie dieses Prinzip, um Alternativen auszuprobieren.
<i>High Cohesion</i> (evaluativ)	Wie kann man Objekte fokussiert, verstehbar und handhabbar halten und nebenbei <i>Low Coupling</i> unterstützen? Weisen Sie Verantwortlichkeiten so zu, dass der Zusammenhang (die Kohäsion) hoch bleibt. Verwenden Sie dieses Prinzip, um Alternativen auszuprobieren.
<i>Polymorphism</i>	Wer ist verantwortlich, wenn das Verhalten je nach Typ variiert? Wenn verwandte Alternativen oder Verhaltensweisen je nach Typ (Klasse) variieren, weisen Sie Verantwortlichkeit für das Verhalten mit polymorphen Operationen den Typen zu, bei denen das Verhalten variiert.
<i>Pure Fabrication</i>	Wer ist verantwortlich, wenn Sie verzweifelt sind und die Prinzipien der <i>High Cohesion</i> und des <i>Low Coupling</i> nicht verletzen wollen? Weisen Sie einen hochkohäsiven Satz von Verantwortlichkeiten einer künstlichen oder geeigneten »Verhaltensklasse« zu, die kein Konzept des Problembereichs repräsentiert, sondern die willkürlich geschaffen wird, um eine hohe Kohäsion, eine geringe Kopplung und die Wiederverwendbarkeit zu unterstützen.
<i>Indirection</i>	Wie kann man Verantwortlichkeiten zuweisen, um eine direkte Kopplung zu vermeiden? Weisen Sie die Verantwortlichkeit einem zwischengeschalteten Objekt zu, das zwischen anderen Komponenten oder Diensten vermittelt, sodass diese nicht direkt gekoppelt sind.
<i>Protected Variations</i>	Wie weist man Verantwortlichkeiten Objekten, Subsystemen und Systemen so zu, dass Variationen oder Instabilitäten in diesen Elementen keinen unerwünschten Einfluss auf andere Elemente haben? Identifizieren Sie die Stellen, an denen Variationen oder Instabilitäten zu erwarten sind; weisen Sie Verantwortlichkeiten zu, um diese Stellen durch eine stabile »Schnittstelle« einzuschließen.