

# LU15b - Umsetzung von Integrationstests



Im Gegensatz zu Unittests wollen wir bei Integrationstests das Zusammenspiel verschiedener Komponenten testen. Daher verzichten wir weitgehend auf simulierte Funktionen und Klassen (Mocks).

## Big-Bang-Integration

Bei dieser Strategie wird das Zusammenspiel aller Komponenten (Module, Funktionen, Klassen) gleichzeitig getestet. Dabei verzichten wir vollständig auf Mocks.

Anhand der Hauptsteuerung suchen wir die Hauptkomponenten unserer Applikation. Zu diesen Komponenten schreiben wir Integrationstests. Dabei werden alle Komponenten, die von den Hauptkomponenten angesprochen werden, automatisch auch getestet.

## Top-Down-Integration

Im Gegensatz zur Big-Bang-Integration testen wir immer nur eine Schicht auf einmal.

## Bottom-Up-Integration

Bei dieser Strategie testen wir zuerst die Integration von Komponenten, die keine weiteren Abhängigkeiten haben.

## Basierend auf Unittests

Diese Strategie ist relativ einfach umzusetzen, wenn schon weitgehend vollständige Unittests existieren.

1. Identifiziere eine Testfunktion (nennen wir sie **test\_A**), die keine Mocks einsetzt.
  - Diese Testfunktion wird in der Regel nur eine Funktion (**function\_A**) testen.
  - Dies sind zu Beginn vor allem Tests von Komponenten, die keine weiteren Komponenten aufrufen.
2. Suche eine Testfunktion (**test\_B**), welche einen Mock dieser Funktion **function\_A** nutzt.
3. Erstelle eine Kopie von **test\_B**, wobei du nun die echte **function\_A** aufrufst.

Diese Schritte werden wiederholt, bis das Zusammenspiel aller Komponenten getestet wird.

## M450-LU15



Marcel Suter

From:  
<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:  
<https://wiki.bzz.ch/modul/m450/learningunits/lu15/umsetzung?rev=1732529762>

Last update: **2024/11/25 11:16**

