

Test doubles and Mocks

1. What are similarities and differences between stubs and mocks?

2. Suppose you are gearing up to test a system that has multiple dependencies:
 - a) language specific libraries and utility methods
 - b) intra-object and intra-class dependencies
 - c) inter-class static dependencies
 - d) inheritance (including interfaces) and composition (including polymorphism)
 - e) file IO (input-output)
 - f) Excel spreadsheet dependencies
 - g) database (say with an SQL -structured query language - database) dependencies
 - h) UI (user interface) dependencies (either desktop app or web app); note that most languages have libraries for different UIs.
 - i) network dependencies (over different protocols, for example, HTTP and HTTPS for web services, FTP for file transfer, SMTP for email, etc)
 - j) other system dependencies

Which of the objects involved in these dependencies are better candidates to stub/mock and why?

3. What are the steps of implementing a mock for an object that interacts with a database (both querying for information and saving to the database or updating the database; you can think of the calls to save to the database/ update the database as similar to sending an invoice via a web service)?

4. Next, suppose we want to save the history of the queries to the database - for each query the timestamp of when it was issued and its content and perform additional tests on the history of queries. What are some additional implementations we need to perform?