

Validating Java Application-Layer Functionality

SOAtest's Jtest Tracer Client (described in [Jtest Tracer Client](#)) can be used to identify, isolate, then reproduce bugs in a multi-layered system.

Tracer allows you to rapidly create realistic functional JUnit test cases that capture the functionality covered by your SOAtest test cases. Using Tracer, you can trace the execution of Java applications at the JVM level (without a need to change any code or to recompile), and in the context of a larger integrated system. As your SOAtest test cases execute, Tracer monitors all the objects that are created, all the data that comes in and goes out.

The trace results are then used to generate contextual JUnit test cases that replay the same actions in isolation, on the developer's desktop, without the need to access all the application dependencies. This means that you can use a single machine to reproduce the behavior of a complicated system during your verification procedure.

Since the generated unit tests directly correlate tests to source code, this improves error identification and diagnosis, and allows developers to run these test cases without having to depend on access to the production environment or set up a complex staging environment. This facilitates collaboration between QA and Development: QA can provide developers traced test sessions with code-level test results, and these tests help developers identify, understand, and resolve the problematic code.

