JPF-Core-X:
Tool Verification Test Results (TVTR)

Contents

Introduction

This document is one of a several exemplar documents prepared as part of a research Case Study whose objective is to simulate a Formal Methods Tool qualification exercise under DO-330. The specific tool considered in this case study is the core module of Java PathFinder (JPF-Core). As with any tool qualification exercise, the qualification is done with respect to a specific version of the tool. Therefore, throughout this document, we refer to our version of JPF-Core as "JPF-Core-X", as described in Section a of the Tool Qualification Plan. This particular document provides a representative example of the "Tool Verification Test Results" (TVTR) for JPF-Core-X, and is written according to the guidelines in DO-330 Section 10.2.5.

Because this is a research study, in which there is no actual qualifying organization and accompanying context, and because the tool under consideration is a research implementation without specific versions, release control, user documentation, or standard configurations, some of the sections of an actual TVTR will not be relevant. This document is thus a Framework for an actual TVTR, organized according to the DO-330 enumeration of required contents. Accordingly, some sections will represent content that is concrete enough to be part of an actual TVTR; some will discuss how a more concrete implementation of this tool might fulfill the required contents; and some will not be applicable for the purposes of this research simulation.

The sections that follow are organized according to sub-parts a) through c) of Section 10.2.5 in DO-330. In each section, we attempt to provide representative content of what should appear in an actual TVTR for a real qualification exercise. In addition, we offer supplemental *meta-level* comments throughout the document.

Discussion

This is how a meta-level comment appears in the text. These comments are meant to provide insight into our process of writing the document, and to suggest interesting or important topics that relate to the qualification of formal methods tools.

a. End-to-End Tests

Discussion

In this case study document, results for only a small portion of the full required set of end-to-end test cases are provided. We limit our test cases to those that are included in the JPF distribution.

a.1. Deadlock - Bounded Buffer 1

Test Results Error Message:

```
====================================================== error #1
gov.nasa.jpf.jvm.NotDeadlockedProperty
deadlock encountered:
  thread BoundedBuffer$Producer:{id:1,name:P1,status:WAITING,priority:5,lockCount
    :1,suspendCount:0}
  thread BoundedBuffer$Producer:{id:2,name:P2,status:WAITING,priority:5,lockCount
    :1,suspendCount:0}
  thread BoundedBuffer$Consumer:{id:3,name:C1,status:WAITING,priority:5,lockCount
    :1,suspendCount:0}
```

Snapshot Message:

```
====================================================== snapshot #1
thread BoundedBuffer$Producer:{id:1,name:P1,status:WAITING,priority:5,lockCount:1,
    suspendCount:0}
  waiting on: BoundedBuffer@146
  call stack:
  at java.lang.Object.wait(Object.java)
  at BoundedBuffer.put(BoundedBuffer.java:55)
  at BoundedBuffer$Producer.run(BoundedBuffer.java:91)

thread BoundedBuffer$Producer:{id:2,name:P2,status:WAITING,priority:5,lockCount:1,
    suspendCount:0}
  waiting on: BoundedBuffer@146
  call stack:
  at java.lang.Object.wait(Object.java)
  at BoundedBuffer.put(BoundedBuffer.java:55)
  at BoundedBuffer$Producer.run(BoundedBuffer.java:91)

thread BoundedBuffer$Consumer:{id:3,name:C1,status:WAITING,priority:5,lockCount:1,
    suspendCount:0}
  waiting on: BoundedBuffer@146
  call stack:
  at java.lang.Object.wait(Object.java)
  at BoundedBuffer.get(BoundedBuffer.java:66)
  at BoundedBuffer$Consumer.run(BoundedBuffer.java:110)
```

Results Message:

```
====================================================== results
error #1: gov.nasa.jpf.jvm.NotDeadlockedProperty "deadlock encountered:   thread
    BoundedBuffer$Produ..."
```

a.2. Deadlock - Bounded Buffer 2

Test Results Error Message:

```
====================================================== error #1
gov.nasa.jpf.jvm.NotDeadlockedProperty
deadlock encountered:
  thread BoundedBuffer$Producer:{id:1,name:P1,status:WAITING,priority:5,lockCount
    :1,suspendCount:0}
  thread BoundedBuffer$Producer:{id:2,name:P2,status:WAITING,priority:5,lockCount
    :1,suspendCount:0}
  thread BoundedBuffer$Producer:{id:3,name:P3,status:WAITING,priority:5,lockCount
    :1,suspendCount:0}
  thread BoundedBuffer$Producer:{id:4,name:P4,status:WAITING,priority:5,lockCount
    :1,suspendCount:0}
  thread
  BoundedBuffer$Consumer:{id:5,name:C1,status:WAITING,priority:5,lockCount:1,
    suspendCount:0}
```

Snapshot Message:

```
====================================================== snapshot #1
thread BoundedBuffer$Producer:{id:1,name:P1,status:WAITING,priority:5,lockCount:1,
    suspendCount:0}
  waiting on: BoundedBuffer@146
  call stack:
  at java.lang.Object.wait(Object.java)
  at BoundedBuffer.put(BoundedBuffer.java:55)
  at BoundedBuffer$Producer.run(BoundedBuffer.java:91)

thread BoundedBuffer$Producer:{id:2,name:P2,status:WAITING,priority:5,lockCount:1,
    suspendCount:0}
  waiting on: BoundedBuffer@146
  call stack:
  at java.lang.Object.wait(Object.java)
  at BoundedBuffer.put(BoundedBuffer.java:55)
  at BoundedBuffer$Producer.run(BoundedBuffer.java:91)

thread BoundedBuffer$Producer:{id:3,name:P3,status:WAITING,priority:5,lockCount:1,
    suspendCount:0}
  waiting on: BoundedBuffer@146
  call stack:
  at java.lang.Object.wait(Object.java)
  at BoundedBuffer.put(BoundedBuffer.java:55)
```

```
  at BoundedBuffer$Producer.run(BoundedBuffer.java:91)

thread BoundedBuffer$Producer:{id:4,name:P4,status:WAITING,priority:5,lockCount:1,
    suspendCount:0}
  waiting on: BoundedBuffer@146
  call stack:
  at java.lang.Object.wait(Object.java)
  at BoundedBuffer.put(BoundedBuffer.java:55)
  at BoundedBuffer$Producer.run(BoundedBuffer.java:91)

thread BoundedBuffer$Consumer:{id:5,name:C1,status:WAITING,priority:5,lockCount:1,
    suspendCount:0}
  waiting on: BoundedBuffer@146
  call stack:
  at java.lang.Object.wait(Object.java)
  at BoundedBuffer.get(BoundedBuffer.java:66)
  at BoundedBuffer$Consumer.run(BoundedBuffer.java:110)
```

Results Message:

```
====================================================== results
error #1: gov.nasa.jpf.jvm.NotDeadlockedProperty "deadlock encountered:   thread
    BoundedBuffer$Produ..."
```

a.3. NO Deadlock - Bounded Buffer 3

Results Results Message:

```
====================================================== results
no errors detected
```

a.4. Deadlock - Remote Agent

Results Error Message:

```
====================================================== error #1
gov.nasa.jpf.jvm.NotDeadlockedProperty
deadlock encountered:
  thread FirstTask:{id:1,name:Thread-1,status:WAITING,priority:5,lockCount:1,
    suspendCount:0}
  thread SecondTask:{id:2,name:Thread-2,status:WAITING,priority:5,lockCount:1,
    suspendCount:0}

```

Results Message:

```
====================================================== results
error #1: gov.nasa.jpf.jvm.NotDeadlockedProperty "deadlock encountered:   thread
    FirstTask:{id:1,nam..."
```

a.5. Race Condition - Racer

Discussion

For this study, we focus our test case examples only on deadlocks. However, in a real qualification effort, additional end-to-end test cases should be provided to include cases where other property violations (e.g. race conditions, assertions and exceptions) are reported when present. Similarly, more cases would be required to show that JPF-Core-X runs to completion without reporting these violations for SUT's that are known to be free of error.

a.6. Assertion

Discussion

Results from test cases to demonstrate the proper detection and reporting of "assertion" property violations would be listed here.

a.7. Uncaught Exception

Discussion

Results from test cases to demonstrate the proper detection and reporting of "uncaught exception" property violations would be listed here.

b. Unit Tests

Discussion

A partial list of required unit tests are outlined in the "Tool Verification Test Cases and Procedures" document. While the distribution of JPF does provide a regression test suite, it does not provide a comprehensive set of unit tests. Because writing a new test suite is beyond the scope of our case study, test results for unit tests are not provided here.

For a real qualification effort, the test results from all unit tests would be reported here.

c. Discrepancies

DO-330

"Test results: The objective is to ensure that the test results are correct and that discrepancies between actual and expected results are explained." [6.1.4.4.c]

DO-330

"Record and track any discrepancies found using the problem reporting process." [10.2.6.d]

Although many tools may not have failed tests, for qualification purposes it is ok to have failed tests, as long it does not affect the soundness of the tool. For example, if JPF reports a deadlock when the expected result was no deadlock, it is indeed a failed test, but such a failure will not affect soundness and hence qualification. Similarly, if JPF ran out of memory and did not report any result, it is also a failed test (since it is not the expected output), but will not hinder qualifying the tool. A discrepancy section in the TVTR with some meta-commentary about what it will include in a real qualification, will satisfy all objectives of 10.2.6

Discussion

It is possible that, within the full set of test results, some cases may not pass. For the purposes of qualification, it is acceptable to have failed tests as long as none of those tests impact the soundess of the tool. For example, consider test obligation #5, which said we ought to include test cases that show the tool does not report false positives. The tool can fail these tests with no impact on its soundness, and so failure of these tests will not prevent qualification.

For a real qulaification effort, all test failures should be reported in this section. If it can be argued that the test failure should not prevent qualification, then that argument should be provided here.

References