STAFF

Software Testing Automation Framework for FIX Protocol

Contents

1. What is STAFF?
2. Features Supported by STAFF
3. Test Script
   – General Structure
   – Order Flow Test Example
   – Execution Results Analysis
4. References

What is STAFF?

STAFF is Software Testing Automation Framework for FIX Protocol

- It is a set of assumptions, concepts, and practices providing support for automated testing of FIX applications.
- STAFF is based on Apache cross-platform Java-based build tool called Ant.
- STAFF comes complete with a library of hundreds of Actions, as Ant-based tool. Each Action is designed to perform a single task, such as running a program or a service, renaming a group of files, creating a FIX connection, sending and receiving FIX messages, verification of received messages and much more.

Features Supported by STAFF

- Create FIX Acceptor
- Create FIX Initiator
- Send FIX message of the specified structure with specific tag values. Tag values can be set as:
  - constant
  - calculated or generated variable
  - value taken from previously received message
- Validate received messages:
  - messages order
  - message structure – tag presence/absence
  - tag values
- Validate data stored to DB
- FIX Protocol versions supported:
  - FIX 4.0 – FIX 4.4
  - FIX 5.0 SP2
  - Custom versions based on FIX 4.0 – FIX 4.4, FIX 5.0 SP2

Test Script - General Structure

STAFF test script is an XML document containing list of Ant Actions, executed in the specified order.

Each Action in the project corresponds to an individual step that you would normally do to accomplish your task.

STAFF comes complete with a library of hundreds of Actions, as Ant-based tool. Each Action is designed to perform a single task, such as running a program or a service, renaming a group of files, creating a FIX connection, sending and receiving FIX messages, verification of received messages and much more.

- It is easy to create and manage test cases and environments. Save and reuse STAFF projects using the same items as the last time it was run, with new items you select or with the built-in or third party items found on the web. You can even run STAFF in distributed environment using remote actions.
- STAFF is based on Apache cross-platform Java-based build tool called Ant. In theory, Ant is kind of like make, without make's wrinkles. With STAFF you can easily integrate your tests into build process.

```
<target name="run_all" >
   <echo>Clearing STAFF log files...</echo>
   <delete dir="${test.log.dir}" failonerror="true" />
   <echo>Done</echo>

   <echo>Creating STAFF log directiry...</echo>
   <mkdir dir="${test.log.dir}" />
   <echo>Done</echo>

   <echo>Creating FIX Acceptor...</echo>
   <fixAcceptor settingsfile="Acceptor.properties"
                    id="acceptor" />

   <echo>Creating FIX Initiator...</echo>
   <fixInitiator  settingsfile="Initiator.properties"
                   id="initiator" />

   <echo>Starting FIX Acceptor...</echo>
   <fixStart refid="acceptor" />
   <sleep seconds="1" />

   <echo>Starting FIX Initiator...</echo>
   <fixStart refid="initiator" />
   <sleep seconds="1" />
   <echo>Done</echo>

</target>
```

Test Script - Order Flow Test Example

1. Create FIX Acceptor
2. Create FIX Initiator
3. Send New Order Single (D) message from Initiator to Acceptor
4. Check received message:
5. Send Execution Report (8) - New message to Initiator
6. Check received message:
7. Send Execution Report (8) - Filled message to Initiator
8. Check received message

Test Script - Order Flow Test Example

```
<?xml version="1.0" encoding="UTF-8"?>
<project name="SimpleOrderFlow.xml" default="all">
<dirname property="test.base.dir" file="${ant.file}" />
<taskdef resource="net/sf/antcontrib/antlib.xml" />
<property name="test.log.dir" value="${test.base.dir}/log" />
<property name="fix_common.settings.dir" value="./settings" />

<sleep seconds="3" />

<!-- ==================================================== -->
<target name="all" depends="run_all, test, stop_all">
</target>
<!-- ==================================================== -->
<!-- =====================Description===================== -->

<description>
                    Simple Order Flow (FIX44):
                    1. Create FIX Acceptor
                    2. Create FIX Initiator
                    3. Send New Order Single (D) message from Initiator to Acceptor
                    4. Check received message
                    5. Send Execution Report (8) - New message to Initiator
                    6. Check received message
                    7. Send Execution Report (8) - Filled message to Initiator
                    8. Check received message
</description>

<!-- ==================================================== -->
```

Test Script - Order Flow Test Example

```
<!-- ====================================================== -->
<!-- ========= Run all components and clean logs ========= -->

<target name="run_all" >

            <echo>Clearing STAFF log files...</echo>
            <delete dir="${test.log.dir}" failonerror="true" />
            <echo>Done</echo>

            <echo>Creating STAFF log directiry...</echo>
            <mkdir dir="${test.log.dir}" />
            <echo>Done</echo>

            <echo>Creating FIX Acceptor...</echo>
            <fixAcceptor settingsfile="${fix_common.settings.dir}/SimpleOrderFlow_Acceptor.properties" id="acceptor" />

    <echo>Creating FIX Initiator...</echo>
            <fixInitiator settingsfile="${fix_common.settings.dir}/SimpleOrderFlow_Initiator.properties" id="initiator" />

            <echo>Starting FIX Acceptor...</echo>
    <fixStart refid="acceptor" />
            <sleep seconds="1" />

            <echo>Starting FIX Initiator...</echo>
    <fixStart refid="initiator" />
            <sleep seconds="1" />
    <echo>Done</echo>

</target>

<!-- ===================================================== -->
```

Test Script - Order Flow Test Example

```
<!-- ==================================================== -->
<!-- ============== Stop all components ================= -->

<target name="stop_all">

                <echo>Stopping FIX Initiator...</echo>
                <fixStop refid="initiator" />
                <sleep seconds="1" />
     <echo>Done</echo>

                <echo>Stopping FIX Acceptor...</echo>
                <fixStop refid="acceptor" />
                <sleep seconds="1" />
     <echo>Done</echo>

</target>

<!-- ==================================================== -->
```

Test Script - Order Flow Test Example

```
<!-- ============================================= -->
<!-- =================Test========================= -->

<target name="test">
            <echo>Waiting for initiator1 to establish connection...</echo>
            <fixWaitForLogin refid="initiator" timeout="30" sender="SNDR" target="TRGT" type="FIX.4.4"/>
            <echo>Done</echo>

            <tstamp>
            <format property="new_order_trn_time" pattern="yyyyMMdd-hh:mm:ss.SSS" locale="en,US" offset="-5" unit="hour" />
            </tstamp>

            <echo>Send New Order Single requests...</echo>
            <fixSend refid="initiator">
                        <fixMessage id="new_order_single">
                                    <field name="MsgType" value="D" />
                                    <field name="HandlInst" value="1" />                        <!-- set tag value as constant -->
                                    <field name="Symbol" value="Symbol" />
                                    <field name="Side" value="1" />
                                    <field name="OrderQty" value="20000" />
                                    <field name="OrdType" value="2" />
                                    <field name="ClOrdID" value="Order#1" />
                                    <field name="Price" value="34.7" />
                                    <field name="ExDestination" value="1" />
                                    <field name="TransactTime" value="${new_order_trn_time}" />   <!-- set tag value as variable -->
                        </fixMessage>
            </fixSend>
            <echo>Done</echo>
```

Test Script - Order Flow Test Example

```
<echo>Waiting for New Order Single message to be received...</echo>
<fixReceive refid="acceptor" timeout="10" >
        <fixMessage id="new_order_single">
                <field name="MsgType"           value="D" />
                <field name="HandlInst" value="1" />
                <field name="Symbol"            value="Symbol" />
                <field name="Side"              value="1" />
                <field name="OrderQty"          value="20000" />
                <field name="OrdType"           value="2" />        <!-- check tag value -->
                <field name="ClOrdID" />                            <!-- check tag presence -->
                <field name="Price"             value="34.7" />
                <field name="ExDestination" value="1" />
        </fixMessage>
</fixReceive>
<echo>Done</echo>

<echo>Get ClOrdID from the message received...</echo>
<fixField refid="new_order_single" action="get" name="ClOrdID" property="ClientOrderID" />
<echo>Done</echo>
```

Test Script - Order Flow Test Example

```
<echo>Create Execution Report message...</echo>
<tstamp>
            <format property="OrderID" pattern="yyyyMMdd-hh:mm:ss.SSS" locale="en,US" offset="-5" unit="hour" />
</tstamp>
<tstamp>
            <format property="ExecID" pattern="yyyyMMdd-hh:mm:ss.SSS" locale="en,US" offset="-5" unit="hour" />
</tstamp>
<fixMessage id="execution_report">
            <field name="MsgType"       value="8" />
            <field name="OrderID"       value="${OrderID}" />
            <field name="ClOrdID"       value="${ClientOrderID}" />
            <field name="ExecID"        value="${ExecID}" />                        <!-- refer tag by name -->
            <field name="150"           value="1" />                                <!-- refer tag by number -->
            <field name="39"            value="1" />
            <field name="Symbol"        value="Symbol" />
            <field name="Side"          value="1" />
            <field name="OrdType"       value="2" />
            <field name="151"           value="20000" />
            <field name="14"            value="0" />
            <field name="6"             value="0.0" />
</fixMessage>
<echo>Done</echo>
```

Test Script - Order Flow Test Example

```
<echo>Send Execution Report (8) - New...</echo>
<fixSend refid="acceptor">
            <fixMessage source="execution_report" />
</fixSend>
<echo>Done</echo>

<echo>Waiting for Execution Report (8) message to be received...</echo>
<fixReceive refid="initiator" timeout="10" >
            <fixMessage source="execution_report" />
</fixReceive>
<echo>Done</echo>

<echo>Updtae Execution Report fields for Fill...</echo>
<tstamp>
            <format property="ExecID_Filled" pattern="yyyyMMdd-hh:mm:ss.SSS" locale="en,US" offset="-5" unit="hour" />
</tstamp>
<fixField refid="execution_report" action="set" name="ExecID" property="ExecID_Filled" />
<fixField refid="execution_report" action="set" name="150" value="F" />
<fixField refid="execution_report" action="set" name="39" value="2" />
<fixField refid="execution_report" action="set" name="151" value="0" />
<fixField refid="execution_report" action="set" name="14" value="20000" />
<fixField refid="execution_report" action="set" name="6" value="25.5" />
```

Test Script - Order Flow Test Example

```
        <echo>Send Execution Report (8) - Filled...</echo>
        <fixSend refid="acceptor">
                <fixMessage source="execution_report" />
        </fixSend>
        <echo>Done</echo>

        <echo>Waiting for Execution Report (8) message to be received...</echo>
        <fixReceive refid="initiator" timeout="10" >
                <fixMessage>
                        <field name="MsgType"         value="8" />
                        <field name="ExecID"          value="${ExecID_Filled}" />
                        <field name="150"             value="F" />
                        <field name="39"              value="2" />
                        <field name="151"             value="0" />
                        <field name="14"              value="20000" />
                        <field name="6"               value="25.5" />
                </fixMessage>
        </fixReceive>
        <echo>Done</echo>
</target>
</project>
```

Test Script - Execution Results Analysis

Simple command is used to launch a test:

```
D:\work\STAFF\doc\SimpleOrderFlow>staff -f SimpleOrderFlow.xml > report.txt
```

Set of FIX logs is created during test scrip execution for each FIX session created .

- FIX.4.4-SNDR-TRGT.body
- FIX.4.4-SNDR-TRGT.header
- FIX.4.4-SNDR-TRGT.seqnums
- FIX.4.4-SNDR-TRGT.session
- FIX.4.4-SNDR-TRGT.event.log
- FIX.4.4-SNDR-TRGT.messages.log

All sent and received messages can be extracted, analyzed and reused for bug reporting and further testing.

Script steps are followed by corresponding messages in the console:

- Console messages are generated automatically for most important actions, such as: send message, receive message, validate message (for unsuccessful comparison only) etc.
- Messages can be specified manually when script is created

Console output can be redirected to file. As result, full execution report for the whole test suite can be created as single text file.

Test Script - Execution Results Analysis

Successful test execution example:

```
Buildfile: D:\work\STAFF\doc\SimpleOrderFlow\SimpleOrderFlow.xml
  [taskdef] Software Testing Automation Framework for FIX version 0.3.7

run_all:
     [echo] Clearing STAFF log files...
   [delete] Deleting directory D:\work\STAFF\doc\SimpleOrderFlow\log
     [echo] Done
     [echo] Creating STAFF log directiry...
    [mkdir] Created dir: D:\work\STAFF\doc\SimpleOrderFlow\log
     [echo] Done
     [echo] Creating FIX Acceptor...
     [echo] Creating FIX Initiator...
     [echo] Starting FIX Acceptor...
     [echo] Starting FIX Initiator...
     [echo] Done
test:
     [echo] Waiting for initiator1 to establish connection...
     [echo] Done
     [echo] Send New Order Single requests...
  [fixSend] Send fix message [8=FIX.4.49=10535=D49=SNDR56=TRGT11=Order#121=138=2000040=244=34.754=155=Symbol60=20121210-08:17:31.069100=110=076].
     [echo] Done
     [echo] Waiting for New Order Single message to be received...
[fixReceive] Receive fix message [8=FIX.4.49=13535=D34=249=SNDR52=20121210-11:17:31.09156=TRGT11=Order#121=138=2000040=244=34.754=155=Symbol60=20121210-
08:17:31.069100=110=235].
     [echo] Done
     [echo] Get ClOrdID from the message received...
     [echo] Done
     [echo] Create Execution Report message...
     [echo] Done
```

Test Script - Execution Results Analysis

```
 [echo] Send Execution Report (8) - New...
  [fixSend] Send fix message [8=FIX.4.49=13435=849=TRGT56=SNDR6=0.011=Order#114=017=20121210-08:17:31.22637=20121210-
08:17:31.22539=140=254=155=Symbol150=1151=2000010=169].
     [echo] Done
     [echo] Waiting for Execution Report (8) message to be received...
[fixReceive] Receive fix message [8=FIX.4.49=16435=834=249=TRGT52=20121210-11:17:31.23456=SNDR6=0.011=Order#114=017=20121210-08:17:31.22637=20121210-
08:17:31.22539=140=254=155=Symbol150=1151=2000010=071].
     [echo] Done
 [echo] Update Execution Report fields for Fill...
     [echo] Send Execution Report (8) - Filled...
  [fixSend] Send fix message [8=FIX.4.49=16535=834=249=TRGT52=20121210-11:17:31.23456=SNDR6=25.511=Order#114=2000017=20121210-08:17:31.36137=20121210-
08:17:31.22539=240=254=155=Symbol150=F151=010=154].
     [echo] Done
     [echo] Waiting for Execution Report (8) message to be received...
[fixReceive] Receive fix message [8=FIX.4.49=16535=834=349=TRGT52=20121210-11:17:31.40456=SNDR6=25.511=Order#114=2000017=20121210-08:17:31.36137=20121210-
08:17:31.22539=240=254=155=Symbol150=F151=010=154].
     [echo] Done

stop_all:
     [echo] Stopping FIX Initiator...
     [echo] Done
     [echo] Stopping FIX Acceptor...
     [echo] Done

all:

BUILD SUCCESSFUL
Total time: 12 seconds
```

Test Script - Execution Results Analysis

Unsuccessful test execution example:

In the example below, some expected values are changed on purpose to be incorrect to show error messages of different types.

```
Buildfile: D:\work\STAFF\doc\SimpleOrderFlow\SimpleOrderFlow_with_error.xml
  [taskdef] Software Testing Automation Framework for FIX version 0.3.7

run_all:
     [echo] Clearing STAFF log files...
   [delete] Deleting directory D:\work\STAFF\doc\SimpleOrderFlow\log
     [echo] Done
     [echo] Creating STAFF log directiry...
    [mkdir] Created dir: D:\work\STAFF\doc\SimpleOrderFlow\log
     [echo] Done
     [echo] Creating FIX Acceptor...
     [echo] Creating FIX Initiator...
     [echo] Starting FIX Acceptor...
     [echo] Starting FIX Initiator...
     [echo] Done

test:
     [echo] Waiting for initiator1 to establish connection...
     [echo] Done
     [echo] Send New Order Single requests...
  [fixSend] Send fix message [8=FIX.4.49=10535=D49=SNDR56=TRGT11=Order#121=138=2000040=244=34.754=155=Symbol60=20121210-09:34:29.077100=110=082].
     [echo] Done
     [echo] Waiting for New Order Single message to be received: - incorrect values of Side and Symbol ...
[fixReceive] Receive fix message [8=FIX.4.49=13535=D34=249=SNDR52=20121210-12:34:29.10056=TRGT11=Order#121=138=2000040=244=34.754=155=Symbol60=20121210-
09:34:29.077100=110=239].
```

Test Script - Execution Results Analysis

```
BUILD FAILED
D:\work\STAFF\doc\SimpleOrderFlow\SimpleOrderFlow_with_error.xml:112:
! Task fixReceive message difference exception Error comparing messages: 8=FIX.4.4☺9=135☺35=D☺34=2☺49=SNDR☺52=20121210-
12:33:26.621☺56=TRGT☺11=Order#1☺21=1☺38=20000☺40=2☺44=34.7☺54=1☺55=Symbol☺60=20121210-
09:33:26.598☺100=1☺10=247☺. Cause:

1) Field [Symbol, Symbol2] != [55, Symbol]  does not equal in received message
2) Field [SymbolSfx, Sfx] does not exist in received message
3) Field [Side, 2] != [54, 1]  does not equal in received message

Total time: 9 seconds
```

References

- Ant documentation: http://ant.apache.org/manual/index.html
- Ant FAQ: http://ant.apache.org/faq.html
- Java documentation: http://java.sun.com/javase/downloads/index.jsp