

# Oracle® Application Testing Suite

Release Notes

Version 9.20

E17387-03

November 2010

---

This document includes updated information for Oracle Application Testing Suite version 9.20. The information in this document is more up-to-date than that in the manuals. Many of the issues outlined in this document will be corrected in upcoming releases. If you have any questions or problems, please contact our support group at <http://www.oracle.com/support/index.html>.

## 1 Contents

This document has the following sections:

- [Section 1, "Contents"](#)
- [Section 2, "New Features and Updates in this Release"](#)
- [Section 3, "System Requirements"](#)
- [Section 4, "Installing the Oracle Application Testing Suite"](#)
- [Section 5, "Backwards Compatibility and Upgrading Scripts"](#)
- [Section 6, "Database Migration"](#)
- [Section 7, "Tutorial and Documentation for the Oracle Application Testing Suite"](#)
- [Section 8, "Known Problems, Limitations and Workarounds"](#)
- [Section 9, "Documentation Accessibility"](#)

## 2 New Features and Updates in this Release

This version includes maintenance improvements to the existing features as well as new features outlined below.

### 2.1 Oracle Application Testing Suite 9.20 - New Features and Updates

- **Oracle Database Testing Accelerator Option** - Enables support for direct database testing using ATS. Users will be able to create synthetic automated test scripts for direct database/SQL testing in OpenScript and also use these scripts for load testing in Oracle Load Testing. Users can also import database transactions from Real Application Testing's Database Replay capture to create test scripts or import from custom SQL Script.
- **Adobe Flex/AMF Load Testing Support** - Enables support for automated load testing of Flex/Flash applications using AMF & HTTP protocols. AMF3/AMF0 is a binary protocol over HTTP used by Flex/Flash applications for server

communication which ATS will now be able to support, in addition to HTTP, for load testing Flex applications.

- **Adobe Flex Functional Testing Support** - Enables support for automated functional testing of Flex applications. Flex utilizes rich GUI components which ATS will now natively recognize for automation & validation of Flex application components. ATS is integrated with the Adobe Test Automation API's which must be compiled with the Flex application in order to capture/replay Flex actions in ATS. ATS supports automated functional testing of Adobe Flex applications created with the Flex 3 SDK and utilizing Flex MX controls. Adobe Flex applications built using Flex 4 SDK and applications utilizing Flex Spark controls are not currently supported at this time.
- **Oracle Forms 11g Support** - OpenScript now supports Oracle Forms 11g applications for both automated functional testing and load testing scripts. ATS supports testing of Web browser-deployed, custom Forms applications in addition to Oracle E-Business Suite Forms applications.
- **Oracle E-Business Suite 11i & R12 Test Starter Kits & Sample Scripts** - New and updated test starter kits for EBS R12 (12.1.1, 12.1.2) and 11i (11.5.10.2). Provides sample functional test scripts and load test script for testing EBS applications.
- **Load Test Scripting Enhancements** - New capture mode for Web load test script recording which results in more compact, intuitive load test scripts and require less manual session data correlation.
- **Databank Enhancements** - New options for parameterizing script inputs including ability to select random record from a databank file and ability to select records from a database query through the OpenScript databanks user interface.
- **Oracle Load Testing Synchronization Points** - Enables setting of Virtual User synchronization points within OpenScript load test scripts, to enable testers to synchronize Oracle Load Testing VUs at specific points in their script during a load test run.
- **Oracle Load Testing Reporting Enhancements** - New, more interactive, flex-based load test graphs for viewing real-time and post-run load test results.
- **Oracle Load Testing Usage Auditing** - New interface for Oracle Load Testing usage tracking and auditing.
- **Oracle Load Testing – Enterprise Manager Database Diagnostics Integration** - Provides Oracle Load Testing users with access to database performance diagnostics during load tests for identifying database performance bottlenecks under load.
- **Oracle Test Manager Reporting Enhancements** - New, more interactive, charts and graphs for viewing Oracle Test Manager reports. Support for including custom test run fields when generating custom reports.
- **Oracle Test Manager TMap Project Templates** - New template for creating Oracle Test Manager projects based on CapGemini Sogeti's TMap test methodology. Includes customized fields, reports and associated data.

## 2.2 Oracle Application Testing Suite 9.20 - Additional Notes

- **Oracle Weblogic Server Support** - Oracle Application Testing Suite is certified with Oracle Weblogic Server as the application server for Oracle Load Testing and Oracle Test Manager. Users of Oracle Load Testing and Oracle Test Manager will receive a free restricted use license of Oracle Weblogic Server for use with these

products. Oracle Weblogic Server is automatically installed and configured with ATS and replaces JBoss Server used in prior versions.

- **Oracle Database 11g Enterprise Edition Support** - Oracle Application Testing Suite is certified with Oracle Database 11g Enterprise Edition as the database repository for Oracle Load Testing and Oracle Test Manager. Users of Oracle Load Testing and Oracle Test Manager will receive a free restricted use license of Oracle Database 11g Enterprise Edition for use with these products. Oracle Application Testing Suite also supports Oracle Database 10g. An Oracle 10g Express Edition (XE) database can be automatically installed and configured with ATS for demo purposes and replaces Microsoft Access used in prior versions.
- **Oracle Functional Testing Legacy Scripting Platform** - The legacy Oracle Functional Testing, Visual Basic-based scripting platform and associated components (such as Job Scheduler) are no longer included with the ATS installer starting with version 9.2. Customers that still require this product should use ATS 9.1 or older versions supported under Oracle's Lifetime Support Policy.
- **Testing HTTPS Sites** - When testing HTTPS sites, Oracle Functional Testing's "Proxy Recorder" feature should only be used to record scripts against trusted Web sites.

### 3 System Requirements

Oracle Application Testing Suite has the following system requirements:

#### 3.1 Oracle Functional Testing/OpenScript

Oracle Functional Testing's OpenScript scripting platform has the following system requirements:

- Operating System (32-bit and 64-bit versions): Windows XP, Windows Vista, Windows 2003, Windows 7, Windows 2008
- Memory: Minimum 1 GB
- System: x86, 32-bit or 64-bit processor, 2.6 GHz or faster
- Disk Space: 4 GB minimum
- Browser: Internet Explorer 6.x, 7.x, 8.x. and Firefox 3.5/3.6

#### 3.2 Oracle Load Testing

Oracle Load Testing has the following system requirements:

- Operating System (32-bit and 64-bit versions): Windows XP, Windows Vista, Windows 2003, Windows 7, Windows 2008
- Memory: Minimum 2 GB.
- System: x86, 32-bit or 64-bit processor, 2.6 GHz or faster
- Disk Space: 10 GB minimum free (at least 3 GB free on the system drive)
- Browser: Internet Explorer 7.x, 8.x, Firefox 3.5/3.6
- Database: Oracle Database 10g or 11g
- Application Server: Oracle WebLogic 11g (10.3.3.0)

- Oracle Load Testing Agent also supported on Linux 32-bit & 64-bit, in addition to Windows 32-bit and 64-bit operating systems listed above

---

---

**Note:** These are minimum requirements only and actual requirements for the Oracle Load Testing Server will vary depending on the size and configuration of your load test. If you are running larger load tests, of greater than 1000 Virtual Users, you should consider deploying Oracle Load Testing on a faster server class machine with additional RAM recommended.

---

---

---

---

**Note:** Oracle recommends running the Oracle Load Testing Server and Agents on separate systems for production load testing. The amount of memory required on the Oracle Load Testing Agent systems may increase based on the number of Virtual Users that will be assigned to run on each Agent. For more information, visit the QA/Testing Technology Center on Oracle Technology Network or contact your Oracle representative.

---

---

### 3.3 Oracle Test Manager

Oracle Test Manager has the following system requirements:

- Operating System (32-bit and 64-bit versions): Windows XP, Windows Vista, Windows 2003, Windows 7, Windows 2008
- Memory: Minimum 2 GB.
- System: x86, 32-bit or 64-bit processor, 2.6 GHz or faster
- Disk Space: 10 GB minimum free (at least 3 GB free on the system drive)
- Browser: Internet Explorer 7.x, 8.x, Firefox 3.5/3.6
- Database: Oracle Database 10g or 11g
- Application Server: Oracle WebLogic 11g (10.3.3.0)

---

---

**Note:** These are minimum requirements only and actual requirements for Oracle Test Manager will vary depending on the quantity of test assets stored and number of concurrent users accessing the application. If you have a large quantity of test cases, requirements or issues stored and/or are deploying Oracle Test Manager for a team greater than 10 users, you should consider deploying Oracle Test Manager on a faster server class machine with additional RAM recommended.

---

---

### 3.4 Additional Notes

The following are additional notes about system requirements:

- The ATS installer can automatically install and configure an Oracle 10g Express Edition (XE) database for use with Oracle Load Testing and Oracle Test Manager. Although this is useful for demos, for production usage switching to Oracle's 11g Enterprise Edition database is highly recommended. Oracle 11g Standard Edition and Oracle 10g EE or SE are also supported.

- System memory requirements do not take into account other processes that may be running and using memory on your system, reducing free uncommitted memory available.
- When installed on the same machine, Oracle Load Testing and Oracle Test Manager will run on the same server and use shared system resources. Oracle Functional Test/OpenScript is a separate application that have its own memory and CPU requirements.
- The ATS products will run in 32-bit compatibility mode when installed on 64-bit operating systems.
- The legacy Oracle Functional Testing, VB-based scripting platform and associated components (such as Job Scheduler) are no longer included with the ATS installer starting with version 9.2. Customers that still require this product should use ATS 9.1 or older versions supported under Oracle's Lifetime Support Policy.

## 4 Installing the Oracle Application Testing Suite

This section explains the installation procedures for the Oracle Application Testing Suite.

---



---

**Caution:** The machines where the OpenScript product is installed should be strictly used for testing. The security features of the browsers have been disabled on this machine to enable recording and playback operations. The browsers cannot be used for secure browsing.

---



---

### 4.1 Installing the Applications

There are two ways that you can install Oracle Application Testing Suite:

From the Oracle Web site:

<http://www.oracle.com/technology/software/products/app-testing/index.html>

1. Select and download the Oracle Application Testing Suite (*oats-version.zip*) from the web site and save it to a temporary directory on your hard disk.
  - *oats-version.zip*: Oracle Application Testing Suite and product documentation PDF files without prerequisites.
  - *oats-upgrade-version.zip*: Oracle Application Testing Suite and Weblogic server. This file can be used to upgrade from Oracle Application Testing Suite version 9.10.
  - *oats-docs-version.zip*: Oracle Application Testing Suite documentation PDF files.
  - *oats-linux-agent-x86-version.zip*: Linux Agent only installation for 32-Bit machines.
  - *oats-linux-agent-x86\_64-version.zip*: Linux Agent only installation for 64-Bit machines.
  - *oats-plus-prereqs-version.zip*: Full installation plus all prerequisites (Oracle Application Testing Suite, Product documentation PDF files, WebLogic Server 11gR1 (10.3.3) developer edition, Oracle Database Express Edition 10.3.2.4,

Oracle Instant Client 11.1.0.7.0, .NET Framework 2.0, Eclipse 3.4.1, Sun JRE 1.6.0\_07, Oracle JRockit 4.0.0.1.6.0.R28)

2. Unzip the downloaded file and run the setup.bat file for Oracle Application Testing Suite.
3. Follow the setup instructions to install the Oracle Application Testing Suite.

During the installation you will be asked to provide a master password. *Remember this password.* The master password specified during installation is used to set all of the following user passwords:

- Oracle Test Manager TM users "default" and "administrator".
- Oracle Lad Testing users "default" and "administrator".
- User "JMSAdmin" used for Oracle Load Testing agent to controller authentication.
- User "oats" for Weblogic Server - this is a Weblogic superuser.
- User "oats-agent" in Weblogic Server for JMS communication between agent and controller - this is *not* a Weblogic superuser.
- Database users "oats", "olt", "otm".
- Database users (XE's defaults) "system" or "sys".

#### 4.1.1 Manually Configuring the Installation

The above installation procedure for Oracle Application Testing Suite setup.bat in oats-plus-prereqs-version.zip installs the applications, database, and Web server. If during the installation procedure you select the **Do nothing, I will configure Oracle Application Testing Suite later (advanced)** option you will need to manually create the database schema.

To manually create the database schema:

1. Open a command window and run the DBConfig utility using this command:

```
<installdir>\jdk\jre\bin\javaw.exe -Djava.library.path=<installdir>\install\lib  
-jar DBConfiguration.jar
```

2. Select **Configuration schema** then click **New** on the toolbar.
3. Select **Create new schema & Tables** then enter your database parameters and click **Save**. Make a note of the OATS schema username (i.e. the user table into which the schema will be created).
4. Select **Oracle Load Testing** then click **New** on the toolbar.
5. Select **Create new schema & Tables** then enter your database parameters and click **Save**. Make a note of the OLT schema username (i.e. the user table into which the schema will be created).
6. Edit <installdir>\bin\config\_ds\_offline.py. Change the line:

```
cmo.setValue('oats')
```

so that the value set is the same as the user noted in step 3.

7. Open a command window and set the environment variables in as follows:  
OATS\_HOME: your install folder  
OATS\_PASS: the 'master password' you entered during installation

OATS\_DBPASS: the password for the OATS schema user noted in step2

OATS\_DBURL: the database URL in the following format:

```
jdbc:oracle:thin:@//<hostname>:<port>/<servicename>
```

8. Run `<installdir>\bin\install.bat`. This will delete your previous OATS domain, if any, and recreate it.
9. Delete the environment variables created, OATS\_PASS, OATS\_DBPASS, etc. in Step 7.

## 4.2 Installing the Oracle Load Testing Remote Agent

Oracle Load Testing allows you to distribute your Virtual Users to run from remote Agent machines. The Oracle Load Testing Server will connect to Oracle Load Testing Agent systems to start and run your Virtual Users on those machines. The Oracle Application Testing Suite Remote Agent is a component of the Oracle Application Testing Suite installation that enables Virtual Users to be distributed to these Agent systems. Users can either install the full Oracle Application Testing Suite installation or just the Remote Agent install component on their Agent machines to enable this functionality.

### 4.2.1 Installing the Remote Agent

To install the Remote agent:

1. Download the Oracle Application Testing Suite Agent .zip file from the download Web site.
2. Unzip the downloaded file and run the install.sh executable.
3. Follow the setup procedure to the **Select Components and Installation Directory** screen.
4. Select to install the full Oracle Application Testing Suite or select just the **Remote Agent** check box and clear all other boxes to install just the Remote Agent software.
5. Verify network access from the Controller workstation to the Agent workstations and configure the Agent Workstations as explained in the following section.

### 4.2.2 Configure Remote Agent Service Login

To specify the user login for the Oracle Load Testing Agent Service, open the Services control panel on the Agent machine and change the login credentials for the "Oracle Load Testing Agent Service". By default, the Oracle Load Testing Agent Service will run under the Local System account.

### 4.2.3 Verify Network Access to Agent Systems

Once you have the Oracle Load Testing Server and Agent software installed on the individual systems, you should verify network access between the Oracle Load Testing Server system and each Remote Agent system. This section provides basic tips and techniques to make sure the Oracle Load Testing Server system can successfully communicate with each Remote Agent system.

- Make sure that you have the Oracle Load Testing Agent software loaded on the Agent system(s) and that it is the same version as the Oracle Application Testing Suite software that is loaded on the Oracle Load Testing Server system. The

systems you plan to use as Agents must have either the Oracle Load Testing Agent software or the full Oracle Application Testing Suite installed to work as agents.

- Make sure you can successfully Ping all of the Agent systems from the Oracle Load Testing Server system. The machine names you use to Ping the systems are the same names that you will specify for the Agent systems in the Oracle Load Testing server. You can also use the IP addresses of the agent systems. If you cannot successfully Ping the Agent systems, contact your network administrator to resolve the issue. If you cannot Ping the Agent systems from the Oracle Load Testing Server system, you will not be able to run the Agents from the server.
- Make sure that the same user is logged in on both the Oracle Load Testing server system and all of the Agent systems. All of the Agent systems must have a user logged in to be controlled by the Oracle Load Testing Server system. You may be able to log in as a different user on the Agent systems as long as the user login has the same administrative privileges as the user logged in on the server system.
- From the server system, try mapping a drive on each of the Agent systems using Windows Explorer. Depending on how your network is setup, the server system may not be allowed to start up processes on the Agent systems. The easiest way around this is to map a drive to the Agent system in order to authenticate with Windows.
- In the Oracle Load Testing server add a Visual Script to the Scenario Profiles list. Enter the machine name or IP address of the Agent system where you want to run the Visual Script into the Systems Manager and select that machine in the Systems field on the Build Scenario tab of Oracle Load Testing.

### 4.3 Installing over beta versions or earlier releases of Oracle Application Testing Suite

If you have a 9.20 beta or Early Access version of the Oracle Application Testing Suite installed, you should un-install the earlier release of the Oracle Application Testing Suite before installing this version.

If you are installing over a previous version of Oracle Application Testing Suite, you may consider making backup copies of the following before installing this version:

- OpenScript scripts in repositories or folders under the installation directory (in *installdir\workspace!* directories or *installdir\directoryname* directories).
- Script files in repositories under the installation directory (in *installdir\workspace* directories). Scripts created in earlier versions of Oracle Application Testing Suite can be used in 9.20 without modification.
- Oracle Load Testing reports database.
- Oracle Test Manager database.

If you are installing or upgrading over a previous version of OpenScript, you should close any open scripts in the earlier version before upgrading.

## 5 Backwards Compatibility and Upgrading Scripts

This section provides information about backwards compatibility of OpenScript scripts and upgrading OpenScript scripts.



## 5.1 Statement of Backwards Compatibility

Scripts created in older versions of OpenScript will always run in new versions of the product without modification from the command-line, Oracle Load Testing, and Oracle Test Manager.

Older OpenScript scripts may not be opened or played back in the newer version of the OpenScript User Interface without upgrading them first. See [Section 5.2, "Upgrading Scripts to the New Release"](#) below. The introduction of Script Assets (in Script Properties) requires pre-version 9.1 scripts to be migrated to the current version if they are to be edited in the OpenScript User Interface.

Previously published script API functions are supported in the latest release. Some published API may be marked as deprecated, but will still work in the new release in order to maintain backwards compatibility.

## 5.2 Upgrading Scripts to the New Release

OpenScript requires that scripts be upgraded to the latest release in order to open them in the OpenScript User Interface. You are not required to upgrade a script to the new version unless you want to open the script in the OpenScript User Interface. Older versions of OpenScript scripts can be run without modification from the command-line, Oracle Load Testing, and Oracle Test Manager.

---

---

**Caution:** Version 9.20 and higher scripts cannot be played back in earlier versions of OpenScript, Oracle Load Testing, and Oracle Test Manager. If you want to maintain pre-version 9.20 scripts, you should make a back up copy of your scripts *before* opening and saving them in version 9.20 or higher. OpenScript automatically migrates any pre-version 9.20 scripts when the script is opened and saved in OpenScript version 9.20 or higher.

---

---

### 5.2.1 Opening Older Scripts in OpenScript

OpenScript automatically prompts you to upgrade older version scripts to the current version whenever the script is opened in the OpenScript User Interface. When opening an older script, you can choose not to open the script and the script will not be upgraded.

When prompted to upgrade a script, if the script depends on any child scripts or function libraries, OpenScript provides an option to upgrade the child scripts or function libraries to the new version also.

Once a script is upgraded to a new release, the script cannot be opened or run using older versions of Oracle Application Testing Suite (OpenScript, Oracle Load Testing, or Oracle Test Manager).

### 5.2.2 Migrating Older Scripts in OpenScript

If you wish to upgrade scripts without opening them individually in OpenScript, you can use the **Migrate Scripts** upgrade option on the **Tools** menu. The Migrate Scripts tool lets you select which scripts to migrate to the current version and find any child scripts that also need to be migrated.

Since version 9.1, scripts that will be run in Oracle Load Testing may not specify absolute paths for their repositories or script assets. However, if your pre-9.1 scripts use absolute paths, you may continue to run the same scripts, unmodified, in the

current version of Oracle Load Testing, without issue. As soon as you upgrade the pre-9.1 scripts to the current version using either the OpenScript User Interface or the Migrate Script tool, the script will not playback in Oracle Load Testing until the absolute paths are changed to relative paths. The Migrate Scripts tool does not migrate absolute paths to relative paths or to repository paths. The absolute paths must be changed in the scripts manually.

### 5.2.3 Copying Older Scripts to New Repositories

For scripts created in releases before 9.1, you should maintain the same rigid, legacy Repository/Workspace folder structure as repositoryLocation/workspace!/script. Changing the repository folder structure within which pre-9.1 scripts exist, such as by adding multiple sub-folders within the repository, or by creating sub-folders without an exclamation mark "!" at the end, may prevent certain pre-9.1 scripts from playing back successfully.

## 5.3 Running Mixed Versions of Scripts

You are advised not to run mixed versions of "job" scripts where a parent script calls child scripts or function libraries. This may happen in cases where you may have 9.2x "parent" scripts that run pre-9.1 "child" scripts or function libraries. Although this configuration has been tested and is supported, the combination of mixed versions scripts may lead to unpredictable results and some confusion as to which scripts are the latest version. In addition, mixed version job scripts may not be able to take advantage of some newer features, such as:

- The ability to visually inspect and add child script functions into a parent script. If pre-9.1 child scripts are not upgraded to the current version, OpenScript will not display their available functions in the user interface options.
- Scripts upgraded to version 9.1 or later no longer require that parent scripts add all child script databanks as their own databanks. If pre-9.1 child scripts are not upgraded to the current version, then parent scripts still must have child script databanks added as their own databanks.

## 5.4 Upgrade Details

When an OpenScript script is upgraded from an older version to the current version, the following changes are applied:

- The modules.properties file is updated to reflect the new version numbers of the modules.
- The META-INF/MANIFEST.MF file may be updated to reflect new bundles required by the newer version of the product.
- If the script being upgraded were created in a pre-9.1 release, then a new assets.xml file is created containing similar data as in the script.xml file. The original script.xml file is now unused, but remains in the script folder for troubleshooting purposes.
- The versions.txt file is replaced by the script.properties file.
- The script .JWG file is updated with the new files.
- The script.java file is not modified.

## 6 Database Migration

A database migration utility is now available for users to enable migration of data from MS Access, SQL Server, and Oracle database into an Oracle database. Documentation for the database migration utility can be found in `<installDir>/bin/DataMigration/help`.

## 7 Tutorial and Documentation for the Oracle Application Testing Suite

The Oracle Application Testing Suite includes product documentation in Adobe Acrobat .PDF format and online help for each application and utility. The documentation includes electronic versions of the Getting Started Guide, OpenScript User's Guide, Oracle Load Testing User's Guide, and Oracle Test Manager User's Guide. The PDF files are installed to the `<installDir>/docs` directory.

You need the Adobe® Acrobat® Reader v4 (or newer) to open and view the documents. You can download the reader from the Adobe web site at <http://get.adobe.com/reader/>.

All of the components of the Oracle Application Testing Suite include comprehensive online Help. You can access the online help using the Help menu options inside of each product.

## 8 Known Problems, Limitations and Workarounds

This section lists known issues and workaround solutions for the components in the Oracle Application Testing Suite.

### 8.1 Oracle Application Testing Suite

The following are know issues and workaround solutions for Oracle Application Testing Suite.

- **Problem:** If the Oracle Application Testing Suite Application Service does not start and the log file `<installDir>\oats\servers\AdminServer\logs\AdminServer.log` contains the following error:

```
weblogic.store.PersistentStoreException: [Store:280073]The file store "WLS_
DIAGNOSTICS" could not be opened because it contained a file with the invalid
version 1. A file of version 2 was expected.
```

**Resolution:** Delete the .DAT file in:

`<installDir>\oats\servers\AdminServer\data\store \diagnostics\` and then restart the server.

The behavior may occur when starting Oracle Application Testing Suite Application Service and the service is stays in starting mode even after the restarting service command window disappears. This condition may happen when the Weblogic (WLS) service is stopped abruptly - either from a machine reboot or the process is stopped for some reason.

- **Problem:** In some cases, the Oracle Universal Installer (OUI) hangs after installing the Oracle Application Testing Suite. If this situation occurs, you can verify the installation by checking the installation log (typically `C:\Program`

Files\Oracle\inventory\logs\installActions-xxxx.log) for a "OATS installed successfully." entry.

**Resolution:** Use the Windows Task Manager, Processes tab to end all running cmd . exe processes and, if necessary, end the OUI application process in Task Manager.

- **Problem:** After you install, you can't login to Oracle Load Testing, Oracle Test Manager, Oracle Administrator, etc.

**Resolution:** During install, you will be prompted for a default password to use in the various OATS products where passwords may be required. It's important that you remember or write down this password as it will be needed to login to these applications (username will typically be "default" or "administrator"). If you don't remember the password, you will have to re-run the installer and establish a new default password. The passwords can be changed and new user accounts added through the Administrator after install.

## 8.2 Oracle Load Testing

The following are known issues and workaround solutions for Oracle Load Testing.

- **Problem:** If your script fails to playback successfully in Oracle Load Testing due to Windows authentication or other user permissions related issues with your application, this may be due to the login authentication of the Oracle Load Testing Agent Service. The Oracle Load Testing Agent Service which runs all the Virtual Users on a given Agent system will by default be installed to run under the Local System account. This may cause issues for certain applications if they block access to this account.

**Resolution:** To address this, open the Oracle Load Testing Agent Service in the Services panel and change the login to run under a specific user with appropriate application permissions.

- When using the SNMP data source, new SNMP MIBs that users specify are uploaded to the Oracle Load Testing Server by default and can be accessed only through the Local Oracle Load Testing Data Collector running on that Server. Remote Oracle Load Testing Data Collectors cannot access these MIBs and users need to copy those over manually.
- **Problem:** To use the JMX Data Source, you may need to manually copy the server-specific class files to the *<installdir>\DataCollector\classes* directory on the Data Collector machine. These class files are required for each J2EE Application supported server. The jar files should be same version as the J2EE Application Server. You should get the class files from your Application Server installation. The list of required jar files is contained in the Oracle Load Testing "agents.properties" file located in *<installdir>\DataCollector\properties*.

**Resolution:** To do JMX Discovery on WebLogic 10.3.3, doing the following:

1. Copy *<installdir>\lib\wlfullclient.jar* to *<installdir>\DataCollector\classes\weblogic*
2. Log in to Oracle Load Testing.
3. Select **Metrics** from the **ServerStats** menu then select **New**.
4. Select the JMX Data Source then click **Discover Counters**.
5. Select the local system and click **Edit**.

6. Select Oracle WebLogic 9.1 as the server (port 8088), username `oats` and the password (the password specified when installing the Oracle Application Testing Suite product).
7. Click **OK**.
8. Click **OK** and the discovery process begins. After a few moments, you will see a full list of the MBeans on the server.

The ServerStats JMX discovery caches counter information. The side effect is that it also caches error messages. For example, failure to copy the correct jar files to data collector's classes directory prior to attempting the discovery would result in an error. When the discovery is attempted after copying the jar files, the discovery still may not work because of the cache. The recommended workaround is to re-start the Oracle Load Testing Server and retry the discovery.

- **Problem:** When bringing up Oracle Load Testing (or Oracle Test Manager) you get the following error reported by the browser: "Unable to connect" or "Can't establish a connection to the server at localhost:8088."

**Resolution:** This error indicates that the OATS Application Service is not running. Start (or restart) the OATS Application Service on the server machine where you installed OATS server. You will also want to make sure that the Oracle Database services are also running (this would be OracleServiceXE and OracleXETNSListener if you used the Oracle XE database).

- **Problem:** When bringing up OLT (or OTM) you get the following error reported by the browser: "Error 404--Not Found"

**Resolution:** This error indicates that the OATS Application Service is unable to connect to the Oracle Database. You will also want to make sure that the Oracle Database services are also running and start/restart them if necessary (this would be OracleServiceXE and OracleXETNSListener if you used the Oracle XE database). Then restart the OATS Application Service.

- The remote ServerStats Data Collector requires all of OATS installed on the remote Data Collector machine (Weblogic Server and Oracle Database).

### 8.3 Oracle Test Manager

The following are known issues and workaround solutions for Oracle Test Manager.

- The icons in the Tree View for Requirements, Tests and Issues are automatically assigned and cannot currently be configured by the Administrator.
- After configuring a new database in the Oracle Test Manager Database Configuration dialog, users must restart the Oracle Application Testing Suite Application Service in order for that database to appear in the Database dropdown list on the Oracle Test Manager login page.
- **Problem:** OpenScript Script Execution in Oracle Test Manager.

**Resolution:** The following additional steps are required in order to run the following types of OpenScript scripts from Oracle Test Manager:

- Siebel Functional
- Oracle Forms Functional
- Web Functional scripts that rely on system input events, such as key press or mouse click

It is necessary to run these scripts using an interactive desktop of a named Windows user account that is always logged in.

1. For Siebel and Oracle Forms, the named user's account must have visited the Siebel or Oracle Forms site at least once to ensure that all necessary ActiveX controls and plug-ins are installed in the named user's browser.
2. On the Oracle Test Manager agent machine that will run the scripts, stop the "Oracle Application Testing Suite Agent Service" and configure it to start manually.
3. On the Oracle Test Manager agent machine, login as the named Windows user account that will run the scripts. From a command prompt, run the following all as one command. The .conf file is an argument for the .exe.

```
C:\OracleATS\agentmanager\bin\AgentManagerService.exe -c  
C:\OracleATS\agentmanager\bin\AgentManagerService.conf
```

where C:\OracleATS is the OATS installed folder.

4. The named user account must remain logged into the system at all times that scripts will be run.
- **Problem:** When bringing up Oracle Load Testing (or Oracle Test Manager) you get the following error reported by the browser: "Unable to connect" or "Can't establish a connection to the server at localhost:8088."

**Resolution:** This error indicates that the OATS Application Service is not running. Start (or restart) the OATS Application Service on the server machine where you installed OATS server. You will also want to make sure that the Oracle Database services are also running (this would be OracleServiceXE and OracleXETNSListener if you used the Oracle XE database).

- **Problem:** When bringing up Oracle Load Testing (or Oracle Test Manager) you get the following error reported by the browser: "Error 404--Not Found"

**Resolution:** This error indicates that the OATS Application Service is unable to connect to the Oracle Database. You will also want to make sure that the Oracle Database services are also running and start/restart them if necessary (this would be OracleServiceXE and OracleXETNSListener if you used the Oracle XE database). Then restart the OATS Application Service.

## 8.4 Oracle Database 10g Express Edition

The following are known issues and workaround solutions for Oracle Database 10g Express Edition.

- **Problem:** In some cases, the Oracle Database 10g Express Edition service does not start completely.

**Resolution:** If you receive a 404 Not Found error when starting Oracle Load Testing or Oracle Test Manager, restart the Oracle database service and the Oracle Application testing service, as follows:

1. Open the Control panel and the Administrative Tools.
2. Open Services.
3. Select the Oracle Application Testing Suite Application Service and click **Stop the service**.
4. Select the OracleServiceXE service and click **Restart the service**.

5. Wait for the OracleServiceXE service Status to indicate "Started".
6. Select the Oracle Application Testing Suite Application Service and click **Start the service**.
7. Wait for the Oracle Application Testing Suite Application Service Status to indicate "Started".
8. Restart Oracle Load Testing or Oracle Test Manager.

## 8.5 Oracle Functional Testing - OpenScript

The following are known issues and workaround solutions for Oracle OpenScript.

- **Problem:** Adobe Flex controls do not record during Adobe Flex functional testing.

---

**Note:** The automation libraries/swc files are required for Flex Functional Testing only. This does not apply for Adobe Flex (AMF) load testing which records at the protocol level.

---

**Resolution:** The Flex application must include the Adobe Flex automation libraries either at compile time or at run time. You need at least `automation.swc` and `automation_agent.swc` from the Adobe <flex builder>\sdks\3.5.0\frameworks\libs folder (3.5.0 is an Adobe sdk version). Also, `automation_dmv.swc` is required for charts, advanceddatagrid and olapdatagrid support. See the Creating Applications for Testing section of the Adobe Flex Data Visualization Developer's Guide for additional information about the tasks required to include the Flex automation libraries.

Creating Applications for Testing:

[http://livedocs.adobe.com/flex/3/html/help.html?content=funct\\_est\\_components2\\_15.html#178953](http://livedocs.adobe.com/flex/3/html/help.html?content=funct_est_components2_15.html#178953)

Adobe Flex Data Visualization Developer's Guide:

[http://livedocs.adobe.com/flex/3/html/help.html?content=funct\\_est\\_components2\\_15.html#178953](http://livedocs.adobe.com/flex/3/html/help.html?content=funct_est_components2_15.html#178953)

The Oracle OpenScript `openscript_agent.swc` file must be included when re-compiling Flex applications. The Flex application must be linked with the OpenScript Flex agent located in <install\_dir>\OpenScript\plugins\oracle.oats.scripting.modules.flexFT\_version\flexagent\openscript\_agent.swc or equivalent.

- **Problem:** Automatic proxy configuration (**Preferences - OpenScript - Record - HTTP Module**) for Firefox may not correctly configure the browser proxy.  
**Resolution:** To record through Firefox, you should configure the browser proxy settings manually (Default: host=localhost, port=7777).
- **Problem:** Certain systems may have trouble launching the OpenScript Help (**Help - Help Contents**).

**Resolution:** This can be resolved by configuring the help to open in an external browser window:

1. From the Tester Perspective, select **View - OpenScript Preferences**.

2. Click the "Clear" icon next to the filter text field in the upper left. Click the Help tree node and check the "Use external browser" box in the Help preferences.
  3. Click **OK** and relaunch OpenScript Help.
- **Problem:** When recording a functional testing script you get the following error: "Cannot connect to browser".  
**Resolution:** The browser did not come up in the required amount of time when you clicked the Record button. This could be because the machine was busy or the ATS OpenScript browser add-ons for IE or Firefox were not installed properly. You can try increasing the browser "Startup timeout" setting under OpenScript Preferences, General...Browser section. If that does not work, run the OpenScript Diagnosis Tool under OpenScript Help. The Diagnosis Tool will attempt to correct any browser add-on installation problems, and report any failures it encounters.
  - **Problem:** When recording a functional testing script, you do not see any actions added to the script after recording.  
**Resolution:** The browser add-on is not capturing events and sending them to OpenScript during recording. Run the OpenScript Diagnosis Tool under OpenScript Help. The Diagnosis Tool will attempt to correct any browser add-on installation problems, and report any failures it encounters. You can also try switching from IE to Firefox to see if that works better.
  - **Problem:** When you record a Web load test script, you do not see any navigations added to the script after recording.  
**Resolution:** OpenScript uses an HTTP proxy server between OpenScript and the target application when you record Web load test scripts. If your browser is accessing the Web via a proxy, you may need to chain that proxy through OpenScript's HTTP recording preferences. OpenScript will attempt to chain the proxy automatically for you using the browser's proxy settings. If you do still have problems recording HTTP navigations, check the "HTTP" Record preferences for network interface and proxy settings.  

Also, if you are recording a load test script against a local Web application running on the same machine as OpenScript (like Weblogic Medical Records), you cannot use "localhost" as the server name in the http request. So in the browser, specify your starting URL as: http://systemname:7011/medrec/index.action instead of http://localhost:7011/medrec/index.action. This has to do with the fact that IE bypasses the proxy for localhost addresses and is not an issue for recording in Firefox.
  - **Problem:** When recording an Oracle EBS/Forms functional testing or load testing script, the browser appears to hang when loading the Oracle Forms applications and you are not able to continue.  
**Resolution:** The browser you are using to record your script may not have the proper JVM version downloaded/installed that is required to run the Oracle Forms applications. Before you start recording Oracle EBS/Forms functional testing or load testing scripts, step through the EBS/Forms transaction once in the browser stand-alone (i.e. while not recording) to make sure you have the proper JVM downloaded/installed and confirm that the Forms applications run properly – then try to re-record your script.
  - **Problem:** When recording a Siebel functional testing or load testing script, the browser appears to hang when loading the Siebel High Interactivity applications and you are not able to continue.



**Resolution:** The browser you are using to record your script may not have the Siebel High Interactivity Framework Active-X components installed that are required to run the Siebel HI applications. Before you start recording Siebel functional testing or load testing scripts, step through the Siebel transaction once in the browser stand-alone (i.e. while not recording) to make sure you have the proper components downloaded/installed and confirm that the Siebel HI applications run properly – then try to re-record your script.

- **Problem:** When recording a Siebel functional testing script, you do not see any Siebel High Interactivity component actions recorded in your script.

**Resolution:** In order to record Siebel functional testing scripts in OpenScript, the Siebel Test Automation Component Automation Services (CAS) must be enabled on Siebel Server before recording. Please review the documentation for information on how to enable Siebel Test Automation for Siebel 7.7, 7.8 and 8.x versions. This is not required for Siebel load testing scripts.

### 8.5.1 Web Functional Test Recording with Firefox Troubleshooting

The following section describes Web Functional Test Recording with Firefox Troubleshooting techniques.

- When using the Firefox browser, you should disable or not install any add-ons into the OpenScript-created Firefox profiles.

## 9 Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at <http://www.oracle.com/accessibility/>.

### Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

### Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

### Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/support/contact.html> or visit <http://www.oracle.com/accessibility/support.html> if you are hearing impaired.

---

Oracle Application Testing Suite Release Notes Version 9.20  
E17387-03

Copyright © 1997, 2010, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.