Oracle® Real User Experience Insight

Accelerator for Oracle E-Business Suite Guide

Release 6.5.0 for Linux x86-64

E17193-02

April 2010

ORACLE®

Oracle Real User Experience Insight Accelerator for Oracle E-Business Suite Guide Release 6.5.0 for Linux x86-64

E17193-02

Primary Author: Paul Coghlan

Contributing Author: Robert Brouwer

Preface

Oracle Real User Experience Insight (RUEI) provides you with powerful analysis of your network and business infrastructure. You can monitor the real-user experience, define Key Performance Indicators (KPIs) and Service Level Agreements (SLAs), and trigger alert notifications for incidents that violate them.

Audience

This document is intended primarily for Administrators responsible for maintaining the RUEI installation. It describes how your RUEI installation can be extended to provide specific support for the monitoring of Oracle E-Business Suite (EBS) applications.

Important

If your monitored Web environment contains EBS applications, is it *strongly* recommended that you make use of this support. It not only saves time in the configuration of your EBS applications within RUEI, considers the specific characteristics of the different frameworks upon which EBS applications are built, and makes these applications more compatible, but also ensures that EBS applications are monitored correctly.

The information provided in this guide is specific to RUEI release 6.5 (or any higher maintenance release). If you upgrade your RUEI installation to a higher version, you will also need to upgrade to the latest version of this package.

The monitoring support provided by this package has been verified against EBS R12. However, it is designed to work equally well with other versions of EBS.

Prerequisites

The Administrator should have firm operational knowledge of their organization's network and application environment. In addition, they should have a good understanding of the EBS architecture. Note that assistance from the EBS administrator or application specialist may also be required.

Before proceeding with the configuration procedure described in this guide, RUEI should already have been successfully placed within your organization's network, and the Initial Setup Wizard run to provide information about the network infrastructure. The procedure to do this is described in the *Oracle Real User Experience Insight Installation Guide*.

The RUEI application is a non-intrusive solution to the monitor of the production environment. By default, the Forms protocol does not send all information relevant for

monitoring over the line. Therefore, the Forms server requires some different configuration settings. To apply these settings, a restart of the Forms server is required.

Oracle Forms Support

Oracle Forms can be configured in two modes: servlet and socket. In servlet mode, a Java servlet (called the Forms Listener servlet) manages the communication between the Forms Java client and the OracleAS Forms services. In socket mode, the desktop clients access the Forms server directly. The RUEI accelerator for E-Business Suites supports both servlet and socket mode. A detailed description of the operation and configuration of Oracle Forms in servlet and socket mode is available at http://metalink.oracle.com/metalink/plsql/ml2_documents.showNOT?p_id=384241.1.

See Appendix A for information about verifying the mode in which Oracle Forms is configured.

Forms Only Customers

The information provided in this guide is relevant to all EBS customers. However, where information is specific to EBS or Forms-only customers, this is highlighted.

Using This Guide

This guide is organized as follows:

- Chapter 1, "Installation and Configuration" describes how to install and configure the RUEI EBS accelerator package for the monitoring of EBS-based applications.
- Chapter 2, "Monitoring and Reporting Considerations" provides additional information about the EBS frameworks relevant to the monitoring of EBS applications in RUEI.
- Appendix A, "Checking Socket and Servlet Mode" presents a description of how to check whether the Oracle Forms server is running in servlet or socket mode.
- Appendix B, "Troubleshooting" highlights the most common problems encountered when installing the RUEI accelerator for Oracle E-Business Suite.
- Appendix C, "Third-Party Licenses" contains licensing information about certain third-party products included with RUEI.

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.

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at `http://www.fcc.gov/cgb/consumerfacts/trs.html`, and a list of phone numbers is available at `http://www.fcc.gov/cgb/dro/trsphonebk.html`.

Related Documents

For more information, see the following documents in the Oracle Real User Experience Insight library:

- *Oracle Real User Experience Insight Installation Guide*.
- *Oracle Real User Experience Insight User's Guide*.
- *Oracle Real User Experience Insight Accelerator for Siebel Guide*.
- *Oracle Real User Experience Insight Accelerator for PeopleSoft Guide*.
- *Oracle Real User Experience Insight Accelerator for JD Edwards EnterpriseOne Guide*.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1 Installation and Configuration

This chapter describes the procedure for installing the EBS accelerator package, and configuring your application definitions within RUEI to enable the accurate monitoring of EBS-based applications.

1.1 Why Use This Accelerator Pack?

This accelerator package for RUEI enables out-of-the-box monitoring of EBS modules. This monitoring supports user session tracking, the discovery of end-user performance issues, and the identification of application issues associated with EBS modules running both the OA and JTT frameworks, as well as Oracle Forms applications running in servlet and socket mode.

This accelerator package automatically discovers all installed EBS modules, and translates network objects to business functions. This facilitates the measurement and monitoring of real-user transactions, from initial query to their commit as part of business transactions. Individual user actions are automatically matched to the correct module, form, or formblock in order to provide contextual analysis. This state-of-the-art monitoring solution supports the creation of KPIs for critical packaged applications, and the analysis of real-user business transactions.

Note: The dependency in previous versions of RUEI on enabling End User Monitoring (for Oracle Forms version 10.1.2 and higher) or Chronos Monitoring (for Oracle Forms version 6i) was removed in RUEI version 6.0. Note that if enabled, database time reporting is available within RUEI.

1.2 Working Within a Forms-Only Environment

The information presented in this guide is relevant to all customers. However, customers working within a Forms-only environment should pay particular attention to the issues highlighted in this section.

In order for RUEI to accurately report on EBS-based applications, it needs information about your production environment. In particular, it needs to map functional areas to reported names. As explained in Section 1.9, "Synchronizing RUEI With the EBS Production Environment", this is done through running the `create_EBS_info.pl` Perl script. Customers within Forms-only environments are also recommended to run this script and upload the generated `.txt` files within a `.zip` file.

Manually Creating Functional Mappings

The `create_EBS_info.pl` script uses a number of EBS database tables to retrieve information about the installation and configuration of your Oracle Forms instance. The exact database tables used are described in Section 2.2, "Database Tables".

However, the APPLSYS.FND_APPLICATION, APPLSYS.FND_APPLICATION_TL, APPLSYS.FND_FORM, APPLSYS.FND_FORM_TL and other tables used by the script do not exist in a Forms-only environment. Therefore, you can either rely on the default (template) mappings provided with RUEI (described later in this section), or you can specify the required mappings by creating the associated `.txt` files manually.

When creating these files manually, the following tab-separated files are required:

- `EBS_formname2details.txt`: specifies a functional description for each form. Each line in the file should have the following format:

  formname{TAB}*form_description*

  For example:

```
ADSAPCRD        Credit Card Expense Transaction Entry
ADSAPPRC        Procurement Card Transaction Entry
ADSCONC         Running Jobs
ADSCONC         Tax Locations
ADSCSCRC        Healthcare CC
ADSMAILI        Mail Information
ADSRSETUP       ADS Repurpose Setup
ADSSOE          Custom Order Entry
ADSSOE          View Person Life Event Information
AKDAPREG        Application Module Parameters Registry
```

- `EBS_formname2appshort.txt`: specifies the short (3-letter) version of the application name of which each form is part. Each line in the file should have the following format:

  formname{TAB}*short_application_name*

  For example:

```
ADSAPCRD        ads
ADSAPPRC        ads
ADSCONC         ads
ADSCSCRC        ads
ADSMAILI        ads
ADSRSETUP       ads
ADSSOE          ads
AKDAPREG        ak
AKDATTRS        ak
AKDFLOWB        ak
```

- `EBS_appsort2appname.txt`: specifies the mapping between the short (3-letter) application name and the full application name. It has the following format:

  short_application_name{TAB}*application_name*

  For example:

```
abm     Activity Based Management (Obsolete)
ad      Applications DBA
ads     Applications Demonstration Services
ads_dev ADS Development
ahl     Complex Maintenance Repair and Overhaul
```

```
ahm     Hosting Manager(Obsolete)
ak      Common Modules-AK
alr     Alert
ame     Approvals Management
amf     Fulfillment Services (Obsolete)
```

Be aware that the created configuration files must be uploaded for each required suite in a `.zip` file. This may only contain non-empty `.txt` files. In addition, all files must be in the root directory. That is, subdirectories are not permitted. It is important that you upload the correct configuration file for the required suite, and that it is based on the actual production environment. The procedure to update the configuration file is described in Section 1.9, "Synchronizing RUEI With the EBS Production Environment".

Relying on the Default (Template) Mapping

If manually creating the required mappings is not practical, you can simply rely on the default (template) mappings already configured within RUEI. While this approach provides an adequate level of reporting, it is subject to the following restrictions:

- *`form_name`*: normally this would be the 8-character technical name translated to a functional description. However, because this is not available, the 8-character technical name is reported instead.
- *`app`*: normally this would be derived from the mapping file that connects the form name with the application. However, because this is not available, the first three letters of the form name are reported instead.
- *`application_name`*: normally this would be derived from the mapping file. However, because this is not available, the *`app`* is reported instead. For example, `eds` instead of `Application Demonstration Services` as shown in Figure 2–1.

Keeping Matching Information Up-to-Date

Because Forms-only environments typically change over time, it is *strongly* recommended that you regularly review your mapping information. Be aware that the above restrictions will also apply to any forms that have been added to your environment since your last ran the `create_EBS_info.pl` script or manually created the mapping files.

Memory Requirements for Forms-Based Environments

Be aware that the monitoring of Forms-based traffic requires significant amounts of memory. For example, the monitoring of 10,000 simultaneous Forms sessions would require approximately 10 GB of Collector memory. Therefore, it is recommended that you deploy the Collector monitoring Forms-based traffic as a remote Collector with at least 16 GB of RAM. Alternatively, if you are using a single-server deployment, the server should have at least 32 GB of RAM.

In addition, it is recommended that you review the level of system memory available to the Collector. For a single-server deployment with 24 GB of RAM, this should be set to 50%, while for a server with 32 GB of RAM, this should be set to 40%. Information about how to increase the amount of available Collector memory is available at `https://metalink2.oracle.com/metalink/plsql/f?p=130:14:7170176407577419410:::::p14_database_id,p14_docid,p14_show_header,p14_show_help,p14_black_frame,p14_font:NOT,762361.1,1,1,1,helvetica`.

1.3 Overview

EBS is based on several frameworks. Because these frameworks allow customers to extend their applications with their own functionality, RUEI requires information about their implementation in order to correctly monitor them.

The EBS monitoring functionality provided with this accelerator package supports all out-of-the-box EBS functionality, as well as some level of customization. However, it is possible that certain EBS customizations may provide unexpected reporting results within RUEI. This mainly concerns the mapping of functional areas to reported names.

In order to facilitate the correct monitoring of EBS-based applications by RUEI, you need to do the following:

1. Install the accelerator package RPMs on the RUEI Reporter system.
2. Verify the scope of monitored traffic.
3. Specify the cookie technology used to track user sessions.
4. Create and configure the EBS suite(s) required for your EBS-based applications.
5. Synchronize the information held within RUEI with the EBS production environment.
6. Verify and evaluate the EBS suite configuration.

Each of these steps are discussed in more detail in the following sections.

1.4 Installing the Package RPMs

Important: The upgrade of an existing Oracle E-Business Suite accelerator package to release 6.5 *must* be performed at the same time as the upgrade of the RUEI system to version 6.5.

Note it is assumed that a working RUEI system has been installed and configured (as described in the *Oracle Real User Experience Insight Installation Guide*), and is fully operational. Install the EBS accelerator package on the RUEI Reporter system using the following commands as the `root` user:

```
cd /root/RUEI/65
./ruei-install.sh suites
```

1.5 Verifying the Scope of Monitoring

Often the EBS software is configured to use a non-standard port, such as 8000. The port on which your EBS installation is running can be found by examining the login URL. This takes the following format:

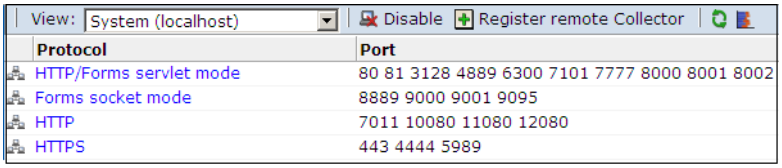
`https(s)://`*`hostname`*`:`***`portnumber`***`/OA_HTML/`*`AppsLogin`*

Verify that the ***`portnumber`*** is configured as one of the defined ports (these are described below). In addition, if a HTTPS port is specified, ensure that a copy of the Web server's private SSL key is imported into the Collector system. See Appendix A, "Checking Socket and Servlet Mode" for information about how to identify the mode and port number.

To verify the port number, do the following:

1. Select **Configuration**, then **Security**, and then **Protocols**. The currently monitored ports are displayed. An example is shown in Figure 1–1:

Figure 1–1 Monitored Ports

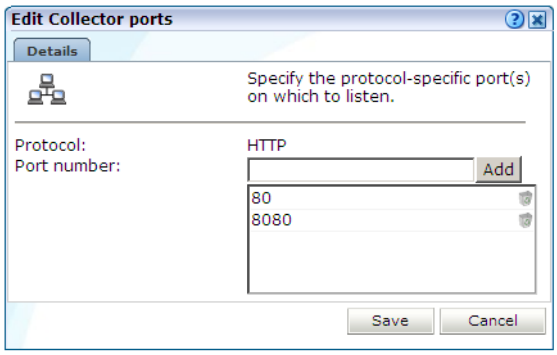


> **Important:** The port numbers specified within each protocol must be mutually exclusive. That is, a port number should only appear in one protocol's list of assigned port numbers.

2. Use the **View** menu to select each Collector. The System (localhost) item represents the local server system.
3. If the port number is not already listed, click the required protocol setting. The following settings are available:
   - **HTTP/Forms servlet mode**: specifies the port(s) on which the Collector should listen for Forms servlet traffic.
   - **Forms socket mode**: specifies the port(s) on which the Collector should listen for Forms traffic in socket mode.
   - **HTTP**: specifies the port(s) on which the Collector should listen for HTTP traffic. This setting should only be used for "pure" HTTP traffic.
   - **HTTPS**: specifies the port(s) on which the Collector should listen for HTTPS traffic.

   A dialog similar to the one shown in Figure 1–2 appears.

Figure 1–2 Edit Collector Ports Dialog



4. To add a new port number, enter the required number in the **Port number** field, and click **Add**. The default Oracle Forms port is 8000. Note, if required, this needs to be manually added. To remove a port from the list, click the **Remove** icon to the right of the port number. When ready, click **Save**.

Important: The port numbers specified within each protocol setting should be mutually exclusive. That is, a port number (such as 8000) should only appear in one protocol's list of assigned port numbers.

5. For each Collector, you are prompted to restart the Collector. This is necessary in order to make your changes effective. Note you can also restart the selected Collector by clicking the **Restart Collector** icon shown in Figure 1–1.
6. For HTTPS traffic, you should also verify that, for each Collector, the required SSL key is installed by selecting **Configuration**, then **Security**, and then **SSL keys**. Use the **View** menu to select the required Collector. If the host name does not match one of the already listed SSL keys, import the required SSL key. The procedure to do this is fully described in the *Oracle Real User Experience Insight User's Guide*. In addition, you should verify that the key is not expired, and activated.

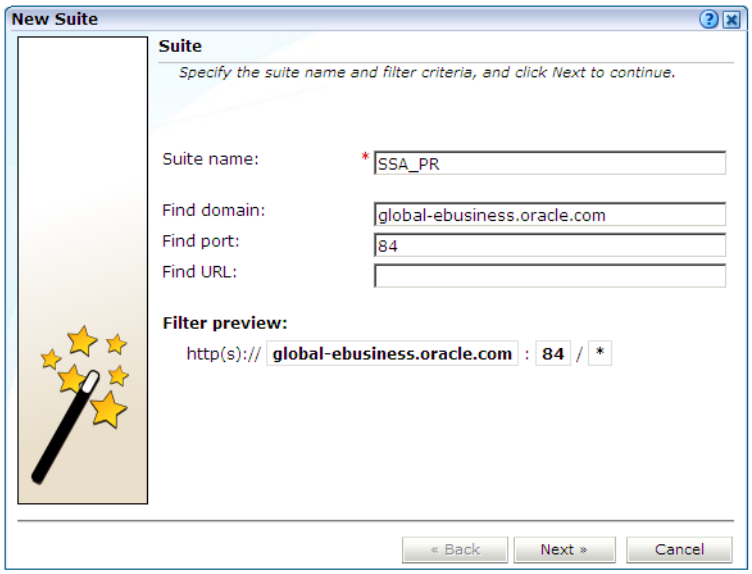
Note: If a port carries HTTPS-based traffic, and also servlet mode traffic, the port should be configured under the **HTTPS** protocol setting (described above).

1.6 Creating the EBS Suite Definition

Within the RUEI Reporter, create and configure the suite definition(s) required for your EBS-based applications. Do the following:

1. Select **Configuration**, then **Applications**, and then **Suites**. Click **New suite**. The dialog shown in Figure 1–3 appears.

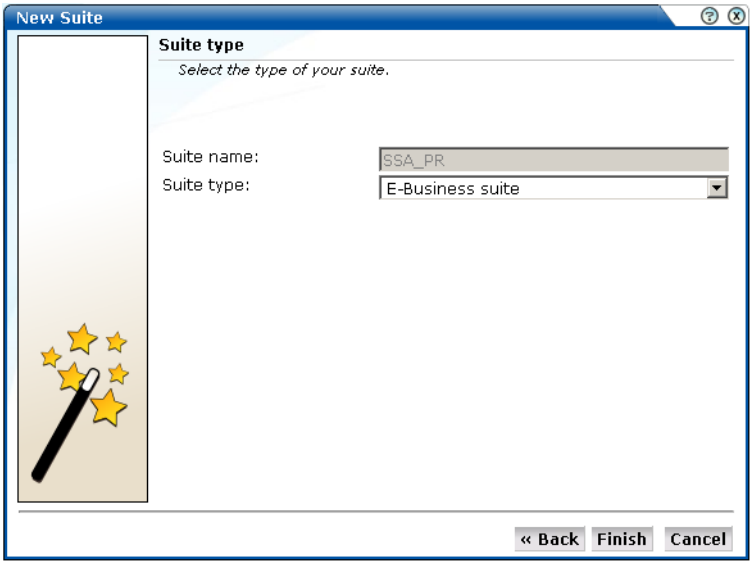
Figure 1–3 New Suite



2. Specify a name for the suite. The name must be unique across suites, services, and applications, and is restricted to a maximum of six characters. Note that the suite cannot be renamed later.
3. Use the remaining fields to specify the scope of the suite. This is defined in terms of partial page URLs. Note that as you enter this information, you can see the

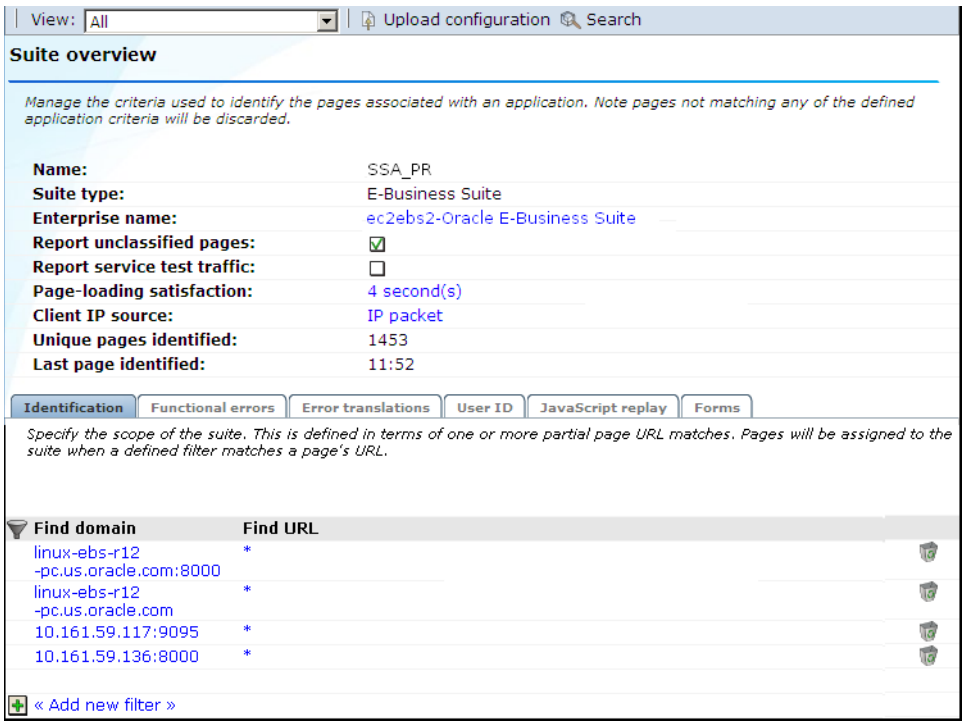
effect of your definition through the **Filter preview** column. Note that if a wildcard character (*) is not specified within the **Domain** field, network traffic arriving on a non-standard port (that is, 80/443), is not associated with the suite. The use of blank filters is not permitted. All specified characters (other than the wildcard character) are interpreted as literals. When ready, click **Next**. The dialog shown in Figure 1–4 appears.

Figure 1–4 Suite Type



4. This dialog allows you to specify the Oracle Enterprise architecture upon which the suite is based. The number of options available in this menu depends on the suite packages currently installed. Select the option **E-Business Suite**. When ready, click **Finish**. The suite definition you have specified is displayed. An example is shown in Figure 1–5.

Figure 1–5 Suite Overview

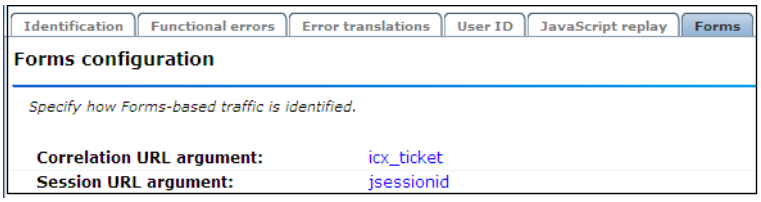


5. If your EBS suite instance uses socket mode, click the **Identification** tab, and use the **Add new filter item** to ensure that the IP addresses and port numbers for all Forms servers are specified. An example is shown in Figure 1–5.

   Note that the filter definition must specify both the IP address and port number, as well as the domain and any URL prefix you wish to specify. In addition, be aware that wildcard characters (*) within IP address and port number definitions are not supported.

6. If your EBS applications make use of Forms, click the **Forms** tab. The suite overview changes to that shown in Figure 1–6 appears.

Figure 1–6 Forms Configuration Tab



The use of these settings is explained in the following section.

1.7 Specifying the Tracking Technology

Within RUEI, session information is based on cookies. The cookies are used to connect hits to a specific visit. In general, the cookie is also connected to the user login page which allows RUEI to include a user name to all subsequent hits with the same cookie.

There are already some cookies available in EBS. However, these are not generally usable. The main problems with them are they not sufficiently unique (for instance, `oracle.uix`), and not wide enough (for instance, JSESSIONID is only used for the `/OA_HTML/` part of the Web site).

It is recommended that you implement a client-side cookie mechanism. The procedure to do so is described in the *Oracle Real User Experience Insight User's Guide*.

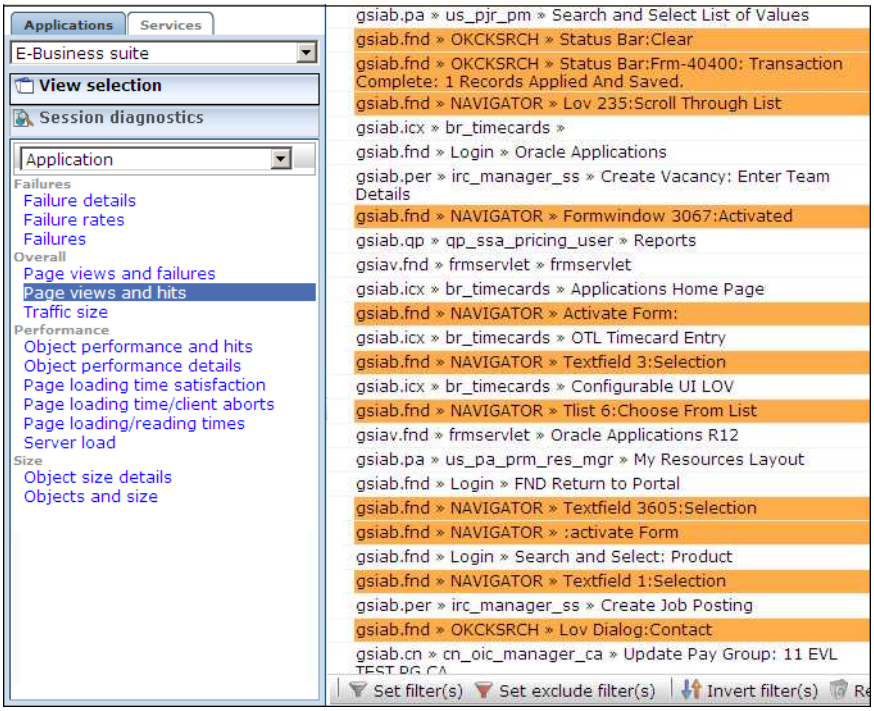
> **Note:** Within a Forms-only environment, if visitors logon to applications *within* Forms, the user ID is automatically tracked on the Forms logon page.

1.7.1 Verifying the Cookie Configuration

To verify your cookie configuration, do the following:

1. Clear all cookies in the browser.
2. (Re)login to the EBS application.
3. Execute some actions that load Oracle Forms.
4. Execute some actions in Oracle Forms.
5. Logout.
6. Wait for at least 10 minutes.
7. Open the RUEI Reporter environment.
8. Select **Browse data**, open the All sessions group, select **Session diagnostics**, and locate the recorded session (by user ID or time). You can filter on applications.
9. Open the session and verify that:
   - There where more page views than just the login page. This verifies that the session ID is preserved in the OA framework after the login.
   - At least some Oracle Forms activity has been recorded with "unidentified action". This verifies that servlet calls are recorded correctly.
   - The page names reported within the Data Browser indicate events similar to those highlighted in Figure 1–7.

Figure 1–7 Example Page Names



When not all hits are connected with the same cookie, it is recommended that you investigate where the problem is located (for instance, the domain or path option of the cookie), and resolve it in the appropriate manner.

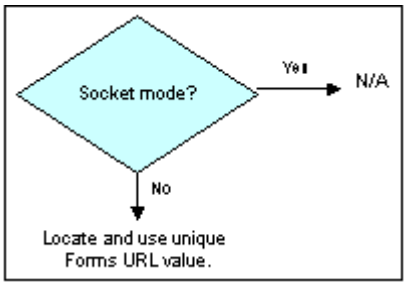
1.7.2 Session Tracking, Correlation Variable, and Session URL argument

The tracking mechanisms that should be specified for the Correlation variable and Session URL argument are best determined through a number of flow charts.

Forms Session URL Argument

Figure 1–8 shows how the Session URL argument can be determined. If running in socket mode, this setting is not applicable. Otherwise, the Forms URL should be examined for an argument that provides a unique value for each Forms session. Typically, this argument is located after a semicolon or question mark character in the URL. For example, `jsessionid` or `JServSessionIdforms`.

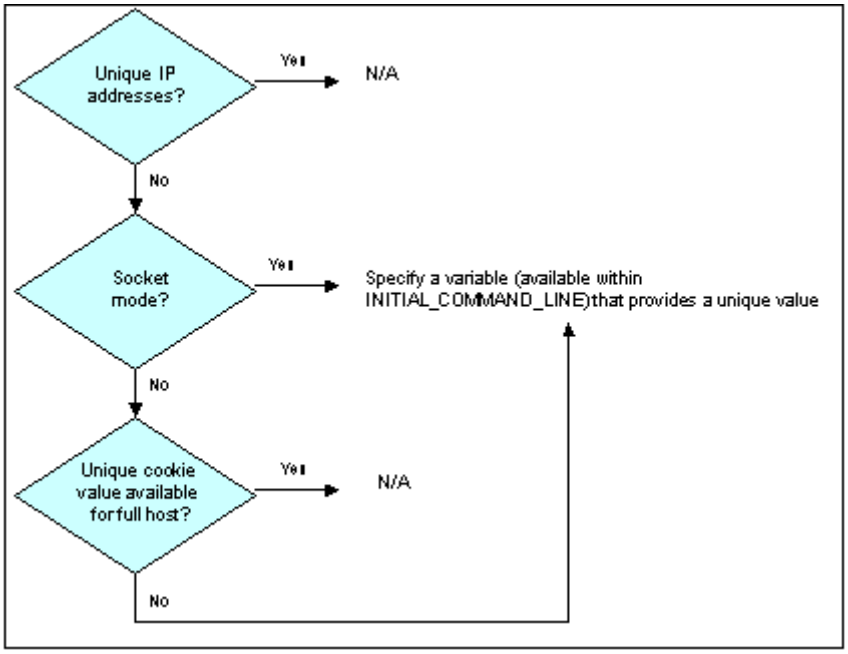
Figure 1–8 Forms Session URL Argument



Correlation Variable

The Correlation variable allows the sessions (on TCP and socket mode) to be merged into one end-user session. Figure 1–9 shows how the Correlation variable can be determined.

Figure 1–9 Correlation Variable



If unique client IP addresses are used, then this setting is not applicable. If running in socket mode, sessions are annotated with the value from the Correlation variable (available via "INDEX_INITIAL_CMDLINE") available on both HTTP and socket-mode traffic. For EBS environments, this will always include the `icx_ticket` variable. For non-EBS environments, some other variable must be specified. On the HTTP layer, the variables are found in the query part of Forms-initializing calls, or in constructions such as *gp1*=. . . .&*gv1*=. . . , where *gp1* specifies the value name.

On the HTTP layer, you might observe the following:

```
/OA_HTML/frmservlet?…&gp15=icx_ticket&gv15=255. 184.210.99jE82BtqiYLHJ8T6-bLxTLw…
```

Alternatively:

```
/OA_HTML/frmservlet?…&env=NLS_LANG='AMERICAN_AMERICA'+…+icx_
ticket='255.184.210.99.jE82BtqiYLHJ8T6-bLxTLw..'+…
```

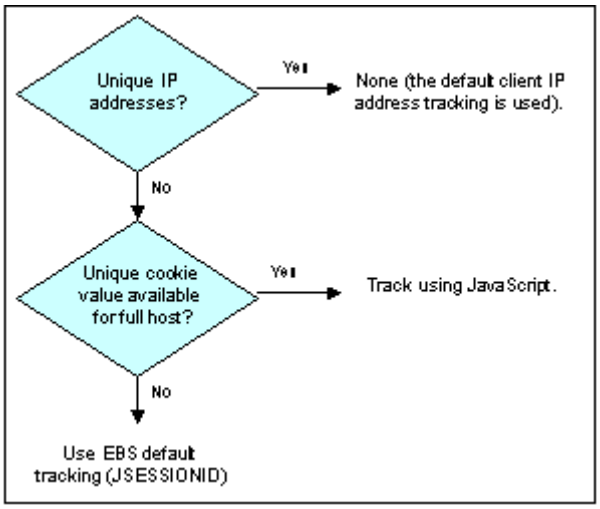
Note that, on the Forms layer, the variable "INDEX_INITIAL_CMDLINE" can be found in the Collector log files. For example:

```
&Runform-001.INDEX_INITIAL_CMDLINE=server module=/oracle/r12/VIS12/apps/apps_
st/appl/fnd/12.0.0/forms/US/FNDSCSGN fndnam=APPS  config='VIS12'
icx_ticket='255.184.210.99.jE82BtqiYLHJ8T6-bLxTLw..' resp='FND/APPLICATION_
DEVELOPER' secgrp='STANDARD' start_func='FND_FNDPOMPO' other_params=…
```

Session Tracking Cookie

Figure 1–10 shows how the session tracking cookie can be determined.

Figure 1–10 Session Tracking Cookie



If unique client IP addresses can be identified, then the default client IP-based tracking can be used. Otherwise, if a cookie with a unique value across the full host is available, then this can be created using JavaScript (as described in the *Oracle Real User Experience Insight User's Guide*). Otherwise, the default EBS (JSESSIONID) tracking scheme should be used.

For example, consider the situation in which it is not possible to modify the login page to add a session cookie. In that case, some other EBS cookie within the non-Forms traffic might be selected (for example, JSESSIONID), and the correlation variable can be used in this case to connect non-Forms traffic with Forms-based traffic. Here, non-Forms hits would be identified using JSESSIONID, shared hits identified by a combination of JSESSIONID and the correlation argument, and Forms hits by the combination of the session-tracking variable `jsessionid` and the correlation argument in the initial command line.

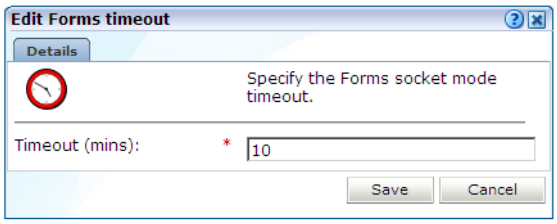
1.8 Specifying The Forms Socket Mode Timeout

The Forms socket mode setting enables you to prevent active socket-mode sessions being discarded by the Collector after they have been inactive for a few minutes. It is recommended that you specify the timeout used within your EBS environment. Note this setting is only relevant for Forms socket mode.

To specify the Forms socket mode timeout, do the following:

1. Select **Configuration**, then **General**, then **Advanced settings**, and then **Collector Forms settings**. Use the **View** menu to select the required Collector. The System (localhost) represents the Collector running on the Reporter system. Click the currently defined Forms socket mode timeout setting. The dialog shown in Figure 1–11 appears.

Figure 1–11 Edit Forms Timeout Dialog



2. Specify (in minutes) the socket mode timeout. The default is 10 minutes. When ready, click **Save**.
3. You are prompted to restart the Collector. This is necessary in order to make your changes effective. Note you can also restart the selected Collector by clicking the **Restart Collector** icon in the toolbar.

Note that you can specify the Forms socket mode timeout to be somewhat higher than the EBS environment timeout. However, be aware that while this has the advantage that sessions are more likely to be successfully detected and monitored, it can increase the amount of required memory.

1.9 Synchronizing RUEI With the EBS Production Environment

In order for RUEI to understand how the EBS frameworks are implemented within your environment, do the following:

1. Copy the `create_EBS_info.pl` script to the home directory of the EBS server. It is located in the `/var/opt/ruei/processor/local/download/ebs` directory of the RUEI system.
2. Run the `create_EBS_info.pl` script as any user on the EBS server[1]. This script assigns an identification to the identified page IDs within the environment. The `create_EBS_info.pl` script must be run with the following syntax:

   ```
   create_EBS_info.pl -part=all|DB|JTT|FORM [-connectstring=connectstring]
   [-debug] [-exeloc=exedir] [-dir=dir1,dir2]
   ```

   where:

   - the `part` option specifies the subset of files to be generated. You can specify the following:
     - `all`: generates all files. This is the default, and is a combination of the three options listed below.
     - `DB`: this option is primarily intended for EBS environments, and generates a subset of the configuration file. If you use this option (or the `all` option), you must specify the `-connectstring` parameter. In addition, you must specify the `-exeloc` parameter. This should specify the location of the SQLPlus executable if it is not in one of the directories in the PATH.
     - `JTT`: this option is primarily intended for EBS environments, and generates all Java-based files. The location of the Java files is based on the APPL_TOP setting. Otherwise, the directories specified with the `-dir` parameter are used.

[1] The script can also be run in the acceptance environment if it is equivalent to the production environment.

  - FORM: this option is primarily intended for Forms-based environments, and generates all Forms-based files. If you specify this option (or the all option), you must specify the -exeloc parameter. This should specify the location of the frmcmp or frmcmp_batch executable if they are not in one of the directories in the PATH. The location of the Forms (.fmb) files is based on the APPL_TOP setting. Otherwise, the directories specified with the -dir parameter are used.

- *connectstring* specifies the string passed to SQLPlus to gain access to the database.
- debug specifies debug mode should be enabled.
- execloc specifies that the executable is not in one of the directories in the PATH, and that the *exedir* directory should be searched. Note that multiple directories must be separated with a comma, or by specifying the -exeloc option multiple times.
- *dir1*, *dir2*, and so on, specify the directories to search for Java or Forms-related information. Note that multiple directories must be separated with a comma, or by specifying the -dir option multiple times.

The script reads from the APPLSYS schema, and generates .txt files in the current directory. For example:

```
perl create_EBS_info.pl -part=all
-connectstring=APPS/APPS@linux-ebs-r12-pc:1522/VIS12
perl create_EBS_info.pl -part=all -connectstring=APPS/APPS@VIS12
```

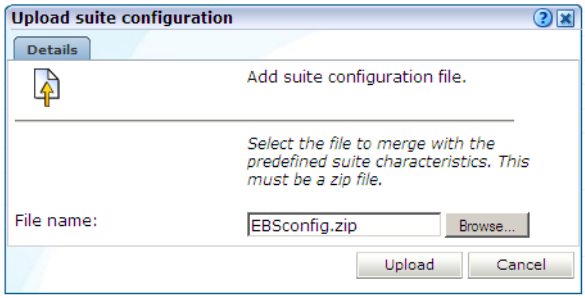
In multiple instance environments, run the script for each required instance, and separately preserve their created .txt files. In addition, create a separate suite definition for each instance, as described in Section 1.6, "Creating the EBS Suite Definition".

Note: If you create new customizations (or make changes to existing customizations) to your EBS applications, you will need to re-run the script, and re-import the generated zip file.

---

3. The script creates a number of .txt files in the directory where the script is executed. All relevant .txt files are collected and stored in a .zip file. Copy this .zip file to a location that can be used for uploading the files to the RUEI Reporter system.
4. Select **Configuration**, then **Applications**, then **Suites**, and select the suite you defined earlier in Section 1.6, "Creating the EBS Suite Definition,". Click **Upload Configuration**. The dialog shown in Figure 1–12 appears.

Figure 1–12 Upload Suite Configuration



5. Specify the name of the `.zip` file containing the generated `.txt` files. If you manually create `.txt` files, you should use the same structure present in the `.zip` file. To protect against receiving empty definitions, the upload will fail when it contains empty `.txt` files. When ready, click **Upload**.

> **Note:** If you receive warning or error messages while running the `create_EBS_info.pl` script, see Section B.7, "create_EBS_info.pl Script Generates Warnings/Errors," for important troubleshooting information.

The Perl Interpreter

By default, the Perl interpreter is not shipped with Microsoft Windows. It is often installed as part of the Oracle database, as well as some other Oracle products. To locate the Perl interpreter on a Microsoft Windows system, select **Start** > **Find** > **Find for files** > `perl.exe`. Use the located executable to run the configuration script.

When no Perl executable is available, you can run the `DB` part of the above query from the RUEI system (providing that a connection to the EBS database from it is possible). This can be achieved by using the `-part=DB` option with a *`connectstring`* that refers to the APPS scheme in the EBS database on the remote host. Note that only the database-based EBS customizations are generated (and not the JTT/Java-based customizations or Forms-based changes).

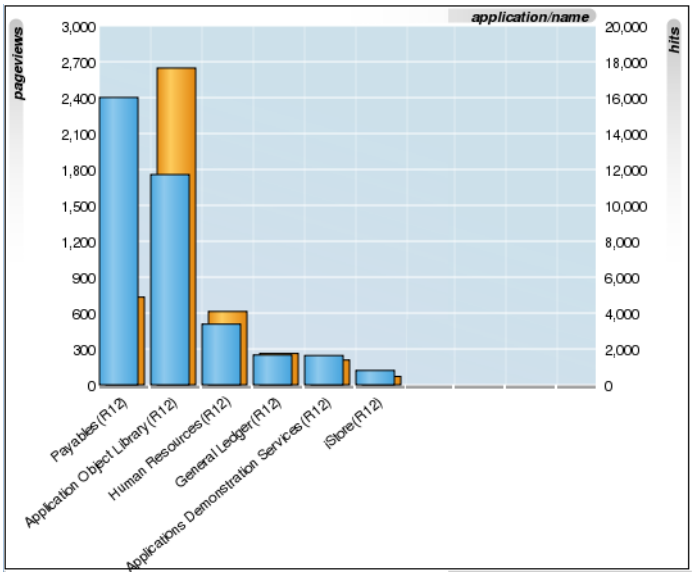
Note that if you skip running the `create_EBS_info.pl` script, RUEI will still report on EBS and Forms activities. However, the reported names will not reflect your customizations. For example, responsibilities will be reported using the responsibility-key instead of the responsibility name, and Forms will be reported using the formname instead of a functional description of the form. This may be acceptable in environments with little customization.

1.10 Verifying and Evaluating Your Configuration

To ensure the quality of the data being collected and reported by RUEI for your EBS-based applications, it is *strongly* recommended that you verify their reported details. You should pay particular attention to the number of associated pages detected for the defined suite(s).

Select **Browse data**, then select the All pages group, and then the Application sub-group. Within the individual dimensions, such as Page views and hits, you can see that page views are reported for several applications. The suite name in the definition is shown between brackets. An example is shown in Figure 1–13.

Figure 1–13 Suite Page Views



Note: The Unique pages identified counter and the Last page identified indicator (shown in Figure 1–5) are disabled. Similarly, the manual page naming facility (described in the *Oracle Real User Experience Insight User's Guide*) is not available.

You can also open an overview of the monitored network traffic by selecting **System**, then **Status**, and then **Data processing**. This provides you with immediate information about hits, pages, and session processing, as well as the system load.

For further information on the user of this and other monitoring facilities, refer to the *Oracle Real User Experience Insight User's Guide*.

1.11 Known Limitations

Currently, RUEI does not work with all EBS functionality. In particular, the following known limitations exist:

- The Forms framework includes functionality to create reports. This functionality is highly configurable by customers. As a result, it is not possible to track reports automatically. In addition, there is no useful translation table with a relevant business-oriented name for the reports. The only solution would to rewrite the known report URLs to correct report names based on a translation file.

  An additional side note on this issue is that some customers are using the 'jobs' functionality to create reports. This is an insecure way to do this, because the next and previous numbers can easily be guessed, and allows users to see reports they may not be authorized to view. Because of the randomness of the name (only a number), it is not useful to report on these type of reports when they are used.

  As a result of the issues described above, Forms reports are not monitored.

- Reporting is based on the last activated area. Hence, when an end-user is browsing simultaneously in multiple browser windows, the reported page name might contain incorrect information.

- Currently, only applications based on the OA and JTT frameworks are supported. Therefore, such packages as Oracle Applications Manager (OAM) and Oracle Portal are not supported at this time.

2

Monitoring and Reporting Considerations

This chapter explains how information within EBS-based applications is captured and reported by RUEI. Information about specific mechanisms or characteristics to be aware of when monitoring EBS-based applications are also highlighted. It is recommended that you review this information to better understand the reporting of EBS applications within RUEI.

2.1 Hostnames and URL Prefixes

An EBS implementation, the EBS instance, can be identified with a hostname and, sometimes, a URL prefix. Generally, an EBS suite can be accessed in two ways: using only the hostname, or using the fully-qualified hostname (including the domain). Generally, you only need to specify the domain, without any specific URL prefix, and the application is accessed at the default location that is configured out-of-the-box.

Table 2–1 shows how an application's dimensions are reported in RUEI.

Table 2–1 EBS Suite Definitions mapping

Dimension level	Content
Application/Name	*application_name* (*suite_name*)
Application/Page group	*suite_name.app* » *form_description*
	suite_name.app » *responsibility_description*
	suite_name.app » *jsp_group*
	suite_name.app » *jsp_name*
	suite_name.app » *servlet_group*
	suite_name.app » *servlet_name*
	suite_name.app » *DAD_location*
Application/Page name	*suite_name.app* » *form_name* » *form_action* » *form_block*
	suite_name.app » *responsibility_key* » *action_description*
	suite_name.app » *jsp_group* » *jsp_name*
	suite_name.app » *jsp_name* » *html_title*
	suite_name.app » *servlet_group* » *servlet_name*
	suite_name.app » *servlet_name* » *html_title*
	suite_name.app » *DAD_location* » *function_name*

Where:

- *action_description* is a description of the action corresponding to one of the following entries in the EBS database:
  - The USER_FUNCTION_NAME column in the FND_FORM_FUNCTIONS_TL table.
  - The ATT_VALUE column in the JDR_ATTRIBUTES table with the property `windowTitle`, `title`, `docName`, or `shortDesc`.
- *application-name* is the name for the application corresponding to the APPLICATION_NAME column in the FND_APPLICATION_TL table.
- *app* is the application short name corresponding to the APPLICATION_SHORT_NAME column in the FND_APPLICATION table.
- *DAD_location* is the location of the pls DAD definition, the full directory, for path that starts with '`/pls/`'.
- *form_action* provides a description of the action, and the element on which the action was performed.
- *form_block* is the name of a functional area within the form.
- *form_description* is the of the form corresponding with the USER_FORM_NAME column in the FND_FORM_TL table.
- *form_name* is the 8-character technical name.
- *function_name* is the function name of the PLS call.
- *html_title* is the title retrieved from the HTML send from the server back to the end user.
- *jsp_group* is the group name assigned to a set of `.jsp` files.
- *jsp_name* is the file name of a `.jsp` file.
- *nservlet_group* is the group name assigned to a set of servlets.
- *nsevlet_name* is the name of an individual servlet.
- *responsibility_key* is the name of the responsibility corresponding with the RESPONSIBILITY_KEY in the FND_RESPONSIBILITY table.
- *suite_name* is the user-defined name specified for the suite upon creation.

Figure 2–1 shows an example of how an EBS application is reported in RUEI.

Figure 2–1 Example of EBS Application Page-Group Reporting

application/name	application/page-group	application/page-name	pageviews
Application Object Library(EBS)	EBS.fnd » NAVIGATOR	EBS.fnd » NAVIGATOR » unidentified action	6128
Application Object Library(EBS)	EBS.fnd » NAVIGATOR	EBS.fnd » NAVIGATOR » Activated:Tlist 93	1966
Application Object Library(EBS)	EBS.fnd » NAVIGATOR	EBS.fnd » NAVIGATOR » Activated:Textfield 89	1838
Application Object Library(EBS)	EBS.fnd » NAVIGATOR	EBS.fnd » NAVIGATOR » Scroll Through List:Lov 2594	1496
Application Object Library(EBS)	EBS.fnd » Enter Assignment	EBS.fnd » PERWSEMA » unidentified action	1481
Application Object Library(EBS)	EBS.fnd » NAVIGATOR	EBS.fnd » NAVIGATOR » Choose From List:Tlist 93	1021
Application Object Library(EBS)	EBS.fnd » Invoice Workbench	EBS.fnd » APXINWKB » unidentified action	839
Application Object Library(EBS)	EBS.fnd » Bills Receivable Transactions	EBS.fnd » ARBRMAIN » unidentified action	774
Application Object Library(EBS)	EBS.fnd » NAVIGATOR	EBS.fnd » NAVIGATOR » Scroll Through List:Lov 3510	671
Application Object Library(EBS)	EBS.fnd » People Management	EBS.fnd » PERWSQHM » unidentified action	640
Application Object Library(EBS)	EBS.fnd » NAVIGATOR	EBS.fnd » NAVIGATOR » Scroll Through List:Lov 437	539
Application Object Library(EBS)	EBS.fnd » XpenseXpress	EBS.fnd » APXXXEER » unidentified action	536
Application Object Library(EBS)	EBS.fnd » People Management	EBS.fnd » PERWSQHM » Pressed:Button 477	495
Application Object Library(EBS)	EBS.fnd » Login	EBS.fnd » FNDSCSGN » unidentified action	453
Application Object Library(EBS)	EBS.fnd » Demand Planning Level Values	EBS.fnd » MSDLVVAL » unidentified action	431
Application Object Library(EBS)	EBS.fnd » NAVIGATOR	EBS.fnd » NAVIGATOR » Choose From List:Tlist 5	411
Application Object Library(EBS)	EBS.fnd » Define Payroll	EBS.fnd » PAYWSDPG » unidentified action	387
Application Object Library(EBS)	EBS.fnd » Enter Assignment	EBS.fnd » PERWSEMA » Scroll Through List:Lov 1058	348

2.2 Database Tables

The following EBS database tables are used by the `create_EBS_info.pl` script to retrieve information about the customizations:

- APPLSYS.FND_FORM_FUNCTIONS

  Function_id, application_id.

  Function_id is used to fill the `EBS_function_id2*.txt` files.

- APPLSYS.FND_FORM_FUNCTIONS

  User_function_name.

- APPLSYS.JDR_PATHS

  Names and the tree structure.

  Path_name is used to fill the `EBS_pathname2*.txt` files.

- APPLSYS.FND_APPLICATION

  Application short name.

  Application_name is used to fill the `EBS_appshort2*.txt` files.

- APPLSYS.FND_APPLICATION_TL

  Application name

- APPLSYS.FND_FORM

  Form_name, application_id

  Form_name is used to fill the `EBS_formname2*.txt` files.

- APPLSYS.FND_FORM_TL

  User-form-name.

- APPLSYS.FND_RESPONSIBILITY

  Responsibility keys

- APPLSYS.FND_RESPONSIBILITY_TL

  Responsibility descriptions

- APPLSYS.JDR_ATTRIBUTES

To make the retrieval easier, the `select` statements make use of the JDR_UTILS and JDR_MDS_INTERNAL packages.

2.3 Actions, Pages, and Objects

Each EBS framework needs to be analyzed to obtain the correct configuration in which all hits are classified as either object hits or action/page hits. Framework-specific considerations are described below.

OA

The OA framework is built using the M-V-C model (Model-View-Controller). Only the controller is relevant to RUEI, because that is the part that will be seen within the HTTP level. The controller decides internally to either show a specific page, or to redirect the visitor to another location that builds up the page. The redirects are recognized automatically; this is normal RUEI functionality.

Based on the URL parameters, the page name is defined (in a redirect situation, the URL of the redirected URL should be used, not the original URL with parameters of the previous page). Besides the controller, the framework also contains some fixed URLs (that by-pass the controller, such as `OALogout.jsp`). These files are recognized together with the JTT-based files.

JTT

The JTT framework is built using the M-V-C model (Model-View-Controller). It differs from the OA framework definition in that there is not one controller for all applications, but one (or multiple) controllers per application. This means that more `.jsp` files are involved, and that requires an investigation of all `.jsp` files involved. A server-side analysis of the `.jsp` files makes it possible to determine the application definition (based on the location of the `.jsp` files).

2.4 Functional Errors

A default RUEI installation recognizes different types of errors. These are in the area of network and HTTP errors. In addition, there is also the facility to manually add functional errors (that is, as site errors). For the EBS frameworks, these content-based errors can be analyzed automatically. To enable this, the functionality described below is implemented.

Oracle Forms Errors

The errors that might occur during a Forms session can be caused by different layers:

- Network errors: are reported in the same way as RUEI does for all applications.
- HTTP server errors (such as 500, 404, and so on) are reported in the same way as all applications are in RUEI.

- Forms servlet errors (servlet connection errors) are reported with their corresponding `ifError` code. These are internal communication errors that occur within the Forms framework.

2.5 OA Framework Page Name Deduction

A detailed discussion of the OA framework is available at http://www-apps.us.oracle.com:1100/fwk/fwksite/510/devguide/ess/ess_state.htm.

OA-based traffic is mapped to RUEI as follows:

- The controller is used as a key indicator for the user-initiated actions. Hits closely related to the controller are assumed to be elements of that page. The OA framework has two controllers: `OA.jsp`, and `RF.jsp`.
- The naming of the page is based on the parameters send to the controller. The following parameters are taken into account: function_id, _rc, akRegionCode, OAFunc, page, and region. Pages that do not contain references to a (new) form or responsibility will preserve the form name or responsibility of previous pages.

Parameter Mapping

Note that the mapping is only possible when the `EBS_*.txt` files are populated with IDs that match the deployments that are being monitored. To obtain the correct configuration files, the script (described in Section 1.9, "Synchronizing RUEI With the EBS Production Environment") is used to retrieve the correct information from the deployment environment.

The script uses two methods to retrieve the relevant information:

- Analysis of local JSP files to obtain the names of all possible JSP files from the JTT environment. This is done through the execution of a `find` statement in the $APPL_TOP directory.
- A list of SQL statements in the `create_EBS_info.pl` script to retrieve the functional names of the OA framework from the database. These are described in the following section.

2.6 Page Context

Not all actions relate to pages. Hence, this section explains how actions (such as HTTP requests) are reported as page views.

Each time a request is received for a page, the OA Framework creates an OAPageContext that persists until a new page finishes processing. Specifically, the OAPageBean, the primary force behind page processing, creates the OAPageContext.

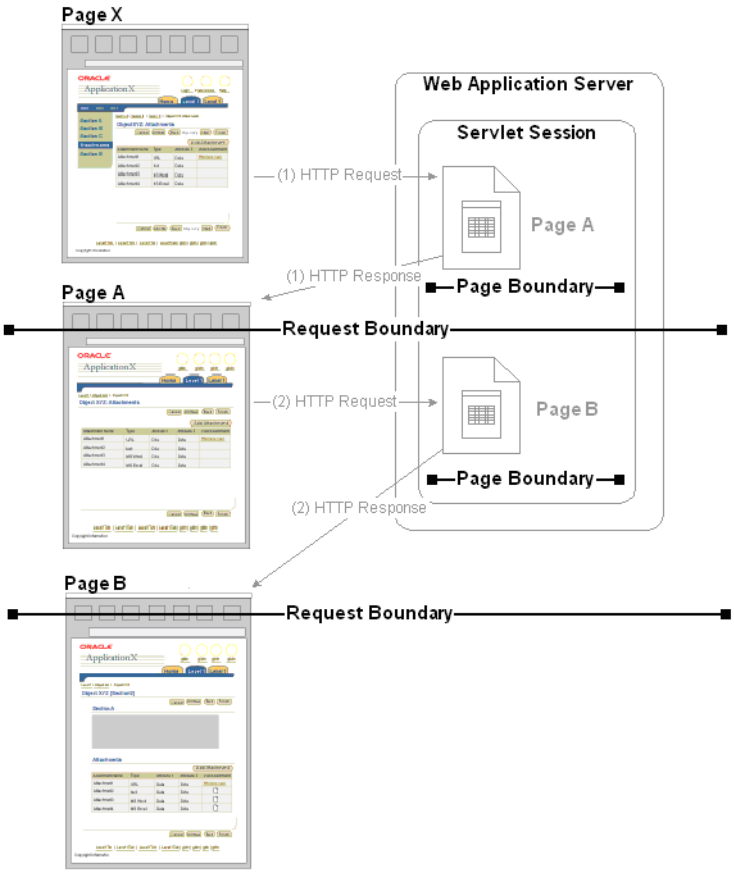
Note that reporting within RUEI is based on the requests seen at the HTTP level. If the page changes within one request, the timings are reported against the original page.

2.6.1 Request and Page Boundaries

A Web application's unit of work is a request/response pair: the browser submits a request, the servlet processes the request, and returns a response. The transmission of a response signifies the end of a single request, or the "boundary" between the completed request and a new one. Similarly, when the OAPageBean finishes processing a page, this is the "boundary" between the current page and a new one.

Hence, in the following scenario where a user navigates from Page X to Page A and then to Page B, we have two request boundaries: the first is between Page X and Page A, and the second is between Page A and Page B. We also have two page boundaries in the same conceptual location between Page X and Page A, and Page A and Page B. This is shown in Figure 2–2.

Figure 2–2 Request and Page Boundaries the same



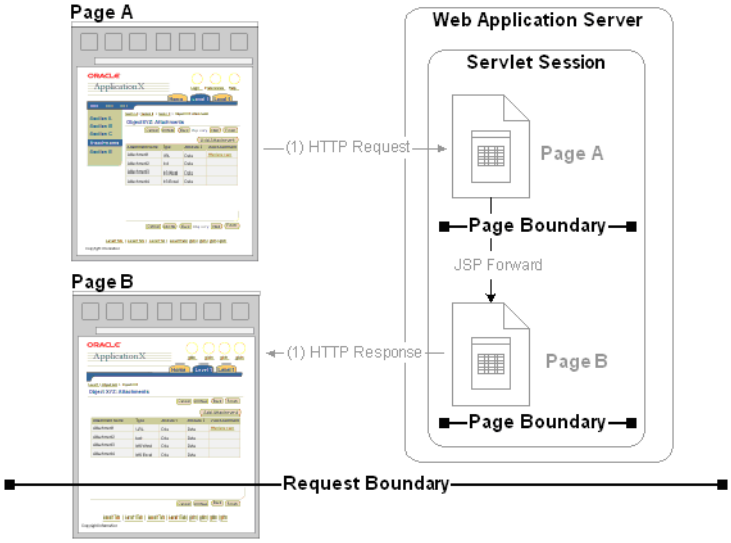
Different Request and Page Boundaries

However, in some situations, the request and page boundaries are not the same. Consider the following JSP Forward case:

- The user navigates from Page X to Page A, as illustrated in Figure 2–2.
- While on Page A, the user selects a control that the Page A code must evaluate before deciding which page to display in response. Therefore, the browser issues a request to Page A which the OA Framework processes (including creating an OAPageContext for the page). Once Page A finishes processing, we've reached the first page boundary as illustrated in Figure 2–3.
- Within the Page A code, the developer evaluates which control the user selected, and issues a JSP Forward to Page B. Instead of providing an HTTP response at this point because we do not want to redisplay Page A, the OA Framework begins processing for Page B (including creating a new OAPageContext for this page). Once Page B finishes processing, we've reached the second page boundary.

- Because Page B must now be displayed to the user, an HTTP response is sent to the browser. We've now reached the request boundary.

Figure 2–3 Different Request and Page Boundaries in the JSP Forward Case



Further information on how a generic JSP application is constructed is available at http://www-apps.us.oracle.com:1100/fwk/fwksite/510/devguide/ess/ess._jspprimer.htm.

2.7 Data Items

The EBS-specific data items shown in Table 2–2 are reported by RUEI.

Table 2–2 EBS-Specific Data Items

Item	Description
EBS suite/Code	The code of an EBS suite, as defined in its configuration definition. This data makes it possible to distinguish between different monitored EBS suites.
EBS suite/Name	The name of an EBS suite, as defined in its configuration definition. This data makes it possible to distinguish between different monitored EBS suites.
EBS framework/Name	The EBS framework used. For example, FORMS (Forms traffic), OA (Oracle Application framework), JTT (JTT framework), servlet (servlets), and other-traffic (only visible when the unclassified pages setting is checked; use page-URL to see the actual URL).
EBS form name/ID	The ID of forms used.
EBS form name/Name	The form description of forms used.
EBS JSP filename/Filename	The name of JSP-based files used. For example, this could contain login-events or actions such as 'runforms'.
EBS responsibility/Key	The responsibility key that was used to access the area. This only applies to OA framework-related URLs, and a limited set of JTT files. In this case, EBS form name reports the form name within which the end user was browsing (using either Forms or the OA framework).

Table 2–2 (Cont.) EBS-Specific Data Items

Item	Description
EBS responsibility/Name	The responsibility description that was used to access the area. This only applies to OA framework-related URLs, and a limited set of JTT files. In this case, EBS form name reports the form name within which the end user was browsing (using either Forms or the OA framework).
EBS module/ID	The ID of the EBS module within which the end user was navigating.
EBS module/Name	The EBS module name within which the end user was navigating.
EBS screen region/ID	The ID of the EBS region within which the end user was navigating.
EBS screen region/Name	The EBS region view within which the end user was navigating.
Total dabase time	The time (in milliseconds) required to execute the Forms-related querries on the database.

2.8 Resources

You may find the information sources useful:

- *Configuring HTTP Server to use SSL in Oracle applications* (note 341904.1).
- *Oracle Forms Service 10g: configuring transport layer security with SSL* (white paper)
- *Oracle Application Server Forms Services Deployment Guide 10g Release 2 (10.1.2), 5.11 Oracle Forms Services and SSL*
- *How to enable SSL for JPI clients (Sun plug-in)* (note 307429.1).

A

Checking Socket and Servlet Mode

This appendix presents a description of how to check whether the Oracle Forms server is running in servlet or socket mode.

Oracle Applications Release 12

Note Oracle Application Release 12 is, by default, configured to run in servlet mode.

Use the following command:

```
$ grep connectMode FORMS_WEB_CONFIG_FILE
```

The current connection mode is reported:

```
connectMode=servlet
```

Alternatively, use the following command:

```
$ grep frmConnectMode CONTEXT_FILE
```

The current connection mode is reported:

```
<forms_connect oa_var="s_frmConnectMode">servlet</forms_conr....
```

Oracle Applications Release 11

Note Oracle Application Release 11 is, by default, configured to run in socket mode.

Use the following command:

```
$ grep connectMode FORMS60_WEB_CONFIG_FILE
```

The current connection mode is reported:

```
connectMode=socket
```

Use the following command:

```
$ grep xsport FORMS60_WEB_CONFIG_FILE
```

The required port number is required:

```
xsport=9095
```

Alternatively, use the following command:

```
$ grep socket CONTEXT_FILE
```

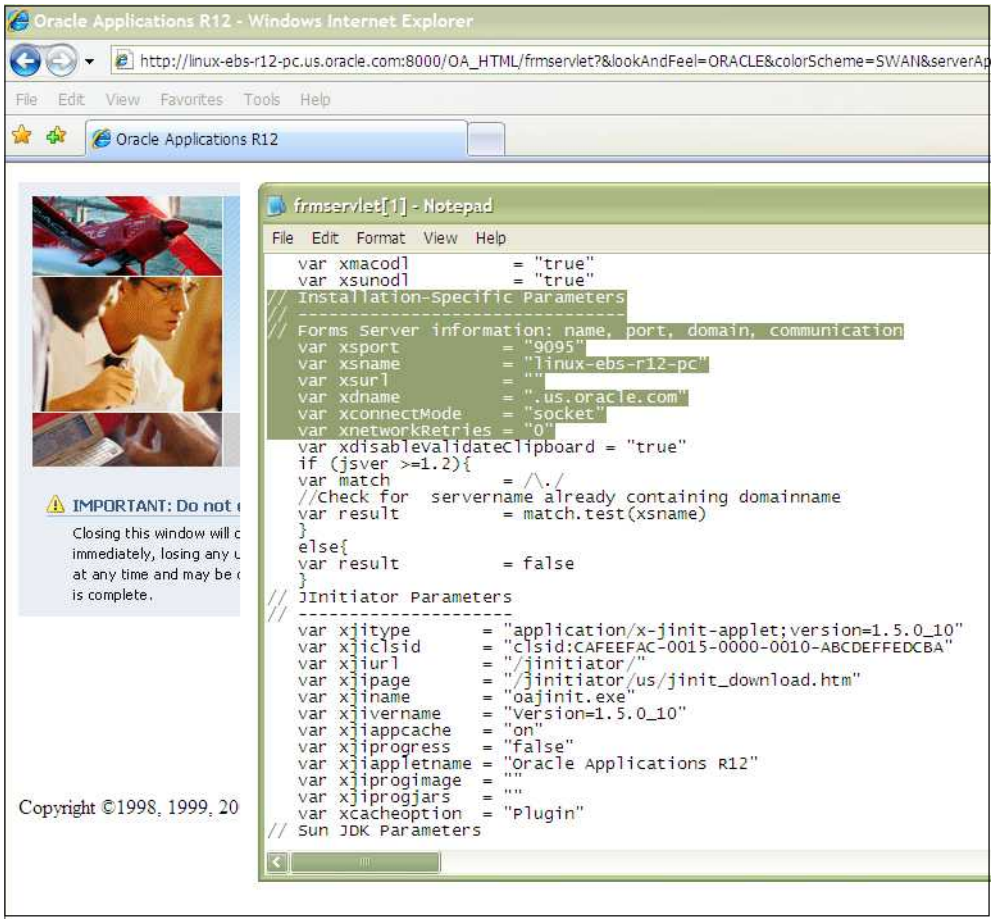
The current connection mode is reported:

```
<forms_connect oa_var="s_frmConnectMode">socket</forms_conr....
```

Checking the HTML Source

Finally, you can also check the HTML source of the page used to launch the Oracle Forms application. To do so within Internet Explorer, select **View**, and then **Source**. This contains the connection mode, as shown in Figure A–1.

Figure A–1 Example Launch Page Details



The relevant connection mode information is highlighted.

B

Troubleshooting

This appendix highlights the most common problems encountered when installing the RUEI accelerator for Oracle E-Business Suite. The information in this appendix should be reviewed before contacting Customer Support.

B.1 Network Traffic Does Not Appear to be Measured

In the event that expected network traffic does not appear to be reported, it is recommended that you review the following points:

- RUEI can monitor EBS applications based on the OA, JTT, PLS, Oracle Forms, and servlet frameworks. Generally, suites are configured to run on a specific port which differs per installation. These also need to be specified in RUEI. Select **Configuration**, then **Security**, and then **Protocols**. Review the defined port settings, and ensure they meet the requirements of your EBS applications.
- Once data starts arriving into the RUEI system, it is not reported automatically. At least one application must be defined. At a minimum, this application must contain the relevant domain name, and the unique page-identification scheme within that domain.
- If the monitored traffic includes VLAN-encapsulated traffic, ensure this is configured within RUEI. Select **System**, then **Configuration**, then **Security**, then **Network filters**, and then **VLAN traffic**, to review the defined settings. The use of this facility is fully described in the *Oracle Real User Experience Insight User's Guide*.
- Be aware that there is no suitable out-of-the-box cookie available for session tracking in EBS. Therefore, a cookie needs to be created on the login page. This should cover the complete application. By default, the `Jession` cookie only covers the application links, and not the images, CGIs, and libraries. While the `oracle.uix` cookie does cover all hits, it is not unique for each visitor.
- Be aware that because the Traffic summary facility (select **System**, then **Status**, and then **Data processing**) is based on application logic, non-application traffic (such as suites, services, and SSOs) is not represented in the traffic overviews.

It is strongly recommended that after installing the EBS accelerator package, you login to the EBS application, and execute a critical path through the application. Then, you should search for recorded action within RUEI, and use the Session Diagnostics facility to verify that it is correctly reported. In particular:

- Verify that descriptions are reported, and not codes. If codes are reported instead of application names, or page-group level codes instead of page-group names, it indicates that the information derived from the `create_EBS_info.pl` script is not activated correctly.

- A large number of reported short sessions indicates that Forms traffic is not being measured.
- A large number of reported `.jsp` files indicates the need for manual page naming (if required by the customer).

B.2 A Large Number of Unidentified Actions are Reported

If a large portion of the reported traffic contains unidentified actions, this indicates that Forms tracking is not functioning correctly. You should consider the following:

- If you do not see such things as "Status Bar" and "Textfield" (as shown in Figure 1–3), this indicates that some specific characteristic in the monitored traffic is not being captured. In this case, you should contact Customer Support.
- If all monitored traffic is reported with unidentified actions, you should verify that the **URL prefix** and **Session URL argument** settings specified within the **Forms** tab of the suite's overview (as shown in Figure 1–6) match those used within your environment. This information is available within the Page URL dimension.
- Verify that the server ports are correctly configured, as described in Section 1.5, "Verifying the Scope of Monitoring". In particular, verify that servlet port is configured as the **HTTP** port.

B.3 Sessions are Reported as "Anonymous"

If sessions are reported as "anonymous", but user IDs are available in the All sessions cube, you should verify the **Correlation URL argument** specified within the **Forms** tab of the suite's overview (as shown in Figure 1–6).

B.4 Create_EBS_info.pl Script Reports FRM-91500 Error

When the `create_EBS_info.pl` script is run on a Unix system, the following error is reported multiple times:

```
FRM-91500: Unable to start/complete the build.
```

This is caused by the `frmbatch` script not having access to the user interface. You should consider the following:

- Ensure that the DISPLAY variable is correctly set. You can use X Window System tools such as `xclock` or `xeyes` to verify it. You might also consider using X-forwarding of SSH to enable the use of the X Windows System on another server.
- The `frmcmp_batch` script is trying to work without the X Windows System. This is the first script used by the `create_EBS_info.pl` configuration script. Set the display mode using the following command:

  ```
  $ set ORACLE_TERM=vt220; export ORACLE_TERM
  ```

B.5 Perl Zip Functionality is not Available

In some systems, zip functionality is not installed as part of the Perl package. In this case, you receive the following message:

```
The Archive::Zip package is not available on this system.
```

After this message, a sample command indicates how the archive might be created. Be aware that the archive should consist of non-empty files, and that files should not be in directories. If so, the upload to RUEI will fail. Alternatively, you can execute the command `zip EBS_*txt` in the appropriate directory.

B.6 The frmcmp_batch Script Fails

The `frmcmp_batch` script fails due to some unknown error, and reports something similar to the following:

```
execution of 'frmcmp_batch module=XXX/XXX/XXX.fmb module_type=form batch=yes
logon=no forms_doc=yes strip_source=yes build=no output_file=/tmp/XXX.txt' failed:
11. Ignoring /XXX/XXX/XXX.fmb
```

This indicates that the reported `.fmb` file could not be converted into `.txt` format (possibly due to corruption). If only a very small proportion of the total number of `.fmb` files are reported, this will probably not be an issue. Indeed, it is likely that the reported forms would not work in a production environment in any case. However, if you know that visitors to your Web site are actively using the reported forms without trouble, then please report this issue. When doing so, please provide the relevant `.fmb` files, together with some indication of how they are deployed within your EBS environment.

B.7 create_EBS_info.pl Script Generates Warnings/Errors

If you receive errors and/or warnings while running the `create_EBS_info.pl` script, depending on their nature, do the following:

- Database related:
  - Verify the *`connectstring`* specified for the `create_EBS_INFO.pl` script by issuing the following command:

    ```
    sqlplus connectstring @temporarysqlfile
    ```

- Forms related:
  - `frmcmp` or `frmcmp_batch` are not working correctly. Detailed troubleshooting information is available about this from Note 266731.1 at https://support.oracle.com/CSP/ui/flash.html.
  - `frmcmp` or `frmcmp_batch` return a sig 11 segmentation fault. This is know to occur for `GRDDHIST.fmb`.

C

Third-Party Licenses

This appendix contains licensing information about certain third-party products included with RUEI 5.1. Unless otherwise specifically noted, all licenses herein are provided for notice purposes only.

The sections in this appendix describe the following third-party licenses:

- Apache Software License, Version 2.0
- OpenSSL
- PHP
- Java Runtime Environment

Apache Software License, Version 2.0

Apache License

Version 2.0, January 2004

`http://www.apache.org/licenses/`

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. **Definitions**. "License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License**. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. **Grant of Patent License**. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. **Redistribution**. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices

that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions**. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks**. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty**. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability**. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability**. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

`http://www.apache.org/licenses/LICENSE-2.0`

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

OpenSSL

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (`http://www.openssl.org/`).

Copyright © 1998-2007 The OpenSSL Project. All rights reserved.

Copyright © 1995-1998 Eric Young (`eay@cryptsoft.com`). All rights reserved.

THIS SOFTWARE IS PROVIDED BY THE OPENSSL PROJECT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OPENSSL PROJECT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA OR PROFITS; OR BUSINESS INTERRUPTION) HOWVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

PHP

Copyright © 1999-2006 The PHP Group. All rights reserved.

This product includes PHP software, freely available from `http://php.net/software/`.

"THIS SOFTWARE IS PROVIDED BY THE PHP DEVELOPMENT TEAM "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE PHP DEVELOPMENT TEAM BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."

Java Runtime Environment

Oracle is required to provide the following notice as part of the license terms for the Sun JRE that the field of use for the JRE is: (i) general purpose desktop computers, laptops and servers, and (ii) embedded systems (by way of example: embedded applications, cell phones, PDAs, TV devices, digital set top boxes, telematics devices and home gateway devices), provided that the Java Runtime Environment (JRE) is licensed only to run Licensee applications, middleware and database products and the JRE is not licensed to directly run any third party applications. This shall not be understood to prevent third party applications from indirectly and incidentally utilizing the JRE, but only as such is required to enable other Licensee Product functionality.

After installing the JRE, the complete terms of the Sun Microsystems license agreement are available in the file `/usr/java/jre1.5.0_18/THIRDPARTYLICENSEREADME.txt`.