

Averify Release Notes

Averify performs an integrity check against the PLM database and reports on specific errors. Averify includes a check for missing indexes and inconsistent data stored in the various database tables. Averify only reports errors, and does not attempt to fix them or modify your database.

Running Averify and using the results to eliminate known errors is a prerequisite for diagnosing problems. Averify is not comprehensive, however, and finding no errors does not certify a database as perfect.

What's New in Averify

- Runs with an Ant-based script, instead of a SQL script.
- Options to check the database at the module, test case, or table level.
- Option to email the log files after Averify runs.
- Ability to use different property files, based on user preferences.
- Ability to specify an NLS language value.

Requirements

- JRE 1.4.2 or 1.5

Downloading Averify

Averify can be downloaded from [Oracle MetaLink](https://metalink.oracle.com) <https://metalink.oracle.com>. The software should be downloaded to your destination database server.

Extract the averify.zip file to a local drive. After Averify is extracted, all of the files are located in the following directories on the local drive:

```
\bin
\config
\lib
\logs
\META-INF
\scripts
```

The averify.properties File

The averify.properties file, located in the \config directory, contains a set of properties where you can specify database details. By default, the file does not contain user-supplied information. You must modify the averify.properties file to complete the destination database details, averify options,

and email options. The destination database parameters contain information about the user and host details of the database you are checking. The Averify options contain information on which portions of the database you want to verify. The email options enable you to send the averify.zip file, containing the averify.log and oracle_averify_report.log, to a specific email address upon completion.

If you do not wish to change the averify.properties file each time you run Averify with different options, you can set the environment variable, PROPERTYFILE, with the full path to another .properties file. You can create multiple .properties files modified with the parameters you need for specific runs of Averify.

Destination Details

Parameter	Value Definition
averify.dest.jdbc.url	The JDBC connection string to the destination database. The format is jdbc:oracle:thin@<DESTINATION_DB_HOST_NAME>:<PORT>:<SID> where DESTINATION_DB_HOST_NAME is the hostname of the destination database, PORT is the listener port of the destination database (1521 is the default), and SID is the database instance.
averify.dest.jdbc.driver	The database driver of the destination database. The default is oracle.jdbc.driver.OracleDriver.
averify.dest.db.user	The database user of the destination database.
averify.dest.db.password	The encrypted database password of the destination database. Use the encryptpwd utility located in the \bin directory to encrypt the password.
averify.dest.tns.name	The TNS entry name in the Oracle Home of the destination database.
averify.dest.oracle.home	The Oracle Home of the destination database. This value can be a mapped drive on the system where Averify is run on the host machine. Note: Use either double back slashes or a forward slash for directory separation, for example, d:/oracle/ora92.

NLS Language Details

Parameter	Value Definition
averify.nls.lang	Specifies the NLS parameter value used by the database. The default value is AMERICAN_AMERICA.UTF8.

Averify Details (For 9.x Databases Only)

Parameter	Value Definition
averify.runOption	Specifies how you want Averify to check the database. The values are all, modules, tables, and testcases. Note: Only one option can be specified for this parameter.
averify.modules	Specific modules chosen for averify to check. This parameter is only used if you entered modules as the value of the runOption parameter. Note: You can check tables that are not specific to the Agile database with the checkNonAgileTables module.
averify.testcases	Specific testcases chosen for Averify to check. This parameter is only used if you entered testcases as the value of the runOption parameter.
averify.tables	Specific database tables chosen for Averify to check. This parameter is only used if you entered tables as the value of the runOption parameter.
<p>An example of how the averify details work is as follows:</p> <pre>averify.runOptions=modules averify.modules=PPM, PCPQM, XXXX averify.testcases=AGIL-0000XXXXXX</pre> <p>Because the specified runOptions parameter is modules, the testcases parameter is ignored when averify is run.</p>	

Email Options

Parameter	Value Definition
averify.email	Decides if the averify.zip file, containing the averify.log and oracle_averify_report.log, is sent as an email attachment upon completion. Value options are true or false. If set to false, the log files can still be viewed in the logs directory.
averify.customer	Name of Agile customer.
averify.mailhost	Name of email server.
averify.mailport	Port number of your email server. The standard default is 25.
averify.toAddress	averify@oracle.com

averify.fromAddress	Email address of the sender.
averify.ccAddress	Email address of an additional recipient.

This section should be configured each time Averify is run. The logs are used by Agile PLM Development only and are not a replacement for logging a request.

Running Averify

Go to the \bin directory and run the averify.bat script on Windows or averify.sh script on UNIX.

If you are running Averify as a Cron job, add the following line to the averify.sh file after the `# Set up the environment` comment line :

```
cd /export/home/oracle/averify/bin
```

Checking the Status of Averify

You can monitor the status through the log file that is generated while Averify is run. The \logs directory contains the averify.log file.

If you chose to have the log files sent as an email attachment, the averify.zip file, containing the averify.log and oracle_averify_report.log, is sent upon completion.