

Automated Testing as a Nightly Process

In this section:

- [Running a Test Suite From the Command Line](#)
- [Running all Projects in a Workspace](#)
- [Running an Individual Project in a Workspace](#)
- [Using a localsettings File](#)
- [Applying an Environment Configuration to a Regression Test from the Command Line](#)

This lesson teaches you how to run tests from the command line, which allows you to configure SOAtest to automatically check the complete project at a specified time each night (or at any other desired interval). This ensures that testing occurs consistently without being disruptive or obtrusive.

SOAtest's command line mode allows you to perform tests from Windows or UNIX command line shells and to run SOAtest from automated build utilities such as Ant, Maven, and CruiseControl.

The following exercises are designed to demonstrate the basics of using `soatestcli`.



Important

A command-line license is required to use `soatestcli`. This license is provided with SOAtest Automation Edition.

Running a Test Suite From the Command Line

In this example, we will run `SOAtestTutorial.tst` from the command line which can be found in the `examples` directory.

1. Close SOAtest and open a command line window.
2. Switch to the directory where SOAtest is installed.
3. From the command line window type the following command:
 - On Windows:

```
soatestcli.exe -config <configuration name> -resource "C:\Location Of SOAtestTutorial.tst (in the default workspace)" -report MySampleReport
```
 - On UNIX (where `Location Of SOAtestTutorial.tst` represents the location of SOAtest on disk):

```
soatestcli -config <configuration name> -resource "/Location Of SOAtestTutorial.tst (in the default workspace)" -report MySampleReport
```

For example:

```
soatestcli.exe -config "user://Example Configuration" -resource "user://Examples/SOAtestTutorial.tst" -report MySampleReport
```

Running all Projects in a Workspace

To run all project in a workspace, use the `-data` option.

```
soatestcli.exe -data "c:\mySOAtestWorkspace" -showdetails -config "user://Example Configuration" -report "c:\mySOAtestReports"
```

The `-data` option specifies the Eclipse workspace location.

The `-showdetails` option prints detailed test progress information.

The `-config` option specifies test configuration.

The `-report` option generates an HTML report.

Running an Individual Project in a Workspace

To run an individual project in a workspace, you must specify the project to be tested with the `-resource` option:

```
soatestcli.exe -data "C:\mySOAtestWorkspace" -resource "MyProject" -exclude "MyProject/somebadtesttoskip.tst" -showdetails -config "user://Example Configuration" -report "C:\mySOAtestReports"
```

The `-exclude` option specifies files to be excluded during testing.

Using a localsettings File

Local settings files can control report settings, Report Center settings, error authorship settings, and Team Server settings. You can create different local settings files for different projects, then use the `-localsettings` option to indicate which file should be used for the current command line test.

Each local settings file must be a simple text file. There are no name or location requirements. Each setting should be entered in a single line.

If a parameter is specified in this file, it will override the related parameter specified from the GUI. If a parameter is not specified in this file, SOAtest will use the parameter specified in the GUI.

```
soatestcli.exe -data "c:\mySOAtestWorkspace" -showdetails -config "user://Example Configuration" -report "c:\mySOAtestReports" -publish -localsettings "c:\mylocalsettings.properties"
```

Here is a *sample* localsettings file (it must be customized for your environment):

```
concerto.reporting=true
dtp.server=dtp.server.com
dtp.port=32323
report.dtp.publish=true
concerto.log_as_nightly=true
report.mail.enabled=true
report.mail.server=smtp.server.com
report.mail.domain=server.com
report.mail.subject=My Nightly Tests
report.mail.cc=myemail@company.com
report.mail.exclude.developers=false
scope.sourcecontrol=true
scope.local=false
soatest.license.use_network=true
soatest.license.network.host=ls.server.com
soatest.license.network.port=2002
soatest.license.network.edition=server_edition
```

Applying an Environment Configuration to a Regression Test from the Command Line

One of the greatest benefits of environments is the ability to rerun the same regression suites from the command line without having to open the SOAtest GUI and modify host or URL settings.

First, setup environments as described in [Running Tests in Different Environments](#).

Next, from the command line, run a command like:

```
soatestcli.exe -config <configuration name> -resource <path to test suite name.tst relative to the workspace> -environment "Default Calculator Environment"
```

Finally, switch environments by using a command like:

```
soatestcli.exe -config <configuration name> -resource <path to test suite name.tst relative to the workspace> -environment "Echo Environment"
```

This will run the same suite with the second environment applied to it.