

Name appclient – launches the Application Client Container and invokes the client application typically packaged in the application JAR file

Synopsis appclient [*client_application_classfile* | -client *client_application_jar*]
[-mainclass *main_class_name* | -name *display_name*]
[-xml *sun-acc.xml file*] [-textauth]
[-targetserver *host[:port][,host[:port]...]*]
[-user *username*] [-passwordfile *password_file*]
[*application-options*]

appclient [*jvm-options*]
[-mainclass *main_class_name* | -name *display_name*]
[-xml *client_config_xml_file*] [-textauth]
[-targetserver *host[:port][,host[:port]...]*]
[-user *username*] [-passwordfile *password_file*]
class-selector [*application-options*]

Description Use the appclient command to launch the Application Client Container and invoke a client application that is typically packaged in an application JAR file. The application client JAR file is specified and created during deployment by the Administration Console or the asadmin deploy command with the --retrieve option. You can also retrieve the client JAR file using the asadmin get-client-stubs command.

The Application Client Container is a set of Java classes, libraries, and other files that are required to execute a first-tier application client program on a Virtual Machine for the Java platform (JVM machine). The Application Client Container communicates with the server using RMI-IIOP.

The client JAR file that is retrieved after deploying an application should be passed with the -client or -jar option when running the appclient utility. The client JAR file name is of the form *app-nameClient.jar*. For multiple application clients in an EAR file, you must use the -mainclass or -name option to specify which client to invoke.

If the application client is a stand-alone module or the only client in an EAR file, the Application Client Container can find the client without using the -mainclass or -name options. If you provide a -mainclass or -name value that does not match what is in the client, the Application Client Container launches the client anyway but issues a warning that the selection did not match the information in the client. The warning also displays what the actual main class and name values are for the client.

Options *jvm-options*
optional; you can set JVM options for the client application. These can be any valid java command options except -client or -jar. JVM options can be intermixed with other appclient command options as long as both types of options appear before the *class-selector*.

client_application_classfile

optional; the file system pathname of the client application .class file. A relative pathname must be relative to the current directory. This class file must contain the `main()` method to be invoked by the Application Client Container.

If you use *client_application_classfile* and the class is dependent on other user classes, you must also set the classpath. You can either use the `-classpath` JVM option in the `appclient` command or set the `CLASSPATH` environment variable. For more information about setting a classpath, see [Setting the Class Path, Oracle Solaris Version](http://java.sun.com/javase/6/docs/technotes/tools/solaris/classpath.html) (<http://java.sun.com/javase/6/docs/technotes/tools/solaris/classpath.html>) or [Setting the Class Path, Windows Version](http://java.sun.com/javase/6/docs/technotes/tools/windows/classpath.html) (<http://java.sun.com/javase/6/docs/technotes/tools/windows/classpath.html>).

`-client`

optional; the name and location for the client JAR file.

`-mainclass`

optional; the full classname of the main client application as specified in the `Main-Class` entry in the `MANIFEST.MF` file. Used for a multiple client applications. By default, uses the class specified in the `client.jar`. For example, `com.sun.test.AppClient`.

`-name`

optional; the display name for the client application. Used for multiple client applications. By default, the display name is specified in the `client.jar` `application-client.xml` file which is identified by the `display-name` attribute.

`-xml`

optional if using the default domain, instance, and name (`sun-acc.xml`), otherwise it is required; identifies the name and location of the client configuration XML file. If not specified, defaults to the `sun-acc.xml` file in the `domain-dir/config` directory.

`-textauth`

optional; used to specify using text format authentication when authentication is needed.

`-targetserver`

optional; a comma-separated list of server specifications for ORB endpoints. Each server specification must include at least the host. Optionally, a server specification can include the port as well. If the port is omitted from a server specification, the default value, `3700`, is used for that host.

`-user`

optional; the user name of the authorized administrative user of the domain administration server.

`-passwordfile`

optional; specifies the name, including the full path, of a file that contains the password entries in a specific format.

The entry for a password must have the `AS_ADMIN_` prefix followed by the password name in uppercase letters, an equals sign and the password.

The valid entries in the file are as follows:

```
AS_ADMIN_PASSWORD=administration-password
AS_ADMIN_MAPPEDPASSWORD=mapped-password
AS_ADMIN_USERPASSWORD=user-password
AS_ADMIN_MASTERPASSWORD=master-password
AS_ADMIN_ALIASPASSWORD=alias-password
```

The password can be specified by one of the following means:

- Through the `-passwordfile` option
- Interactively at the command prompt

For security reasons, a password that is specified as an environment variable is not read by the `appclient` utility.

If the `AS_ADMIN_PASSWORD` environment variable has been exported to the global environment, specifying the `-passwordfile` option produces a warning about using the `-password` option. To avoid this warning, unset the `AS_ADMIN_PASSWORD` environment variable.

The master password is not propagated on the command line or an environment variable, but can be specified in the file that the `-passwordfile` option specifies.

The default value for `AS_ADMIN_MASTERPASSWORD` is `changeit`.

class-selector

required; you must specify the client application class using one of the following class selectors.

-jar jar-file

the name and location of the client JAR file. The application client JAR file is specified and created during deployment by the `asadmin deploy` command. If specified, the `-classpath` setting is ignored in deference to the `Class-Path` setting in the client JAR file's manifest.

class-name

the fully qualified class name of the main client application `main()` method to be invoked by the Application Client Container. For example, `com.sun.test.AppClient`.

If you use *class-name* as the class selector, you must also set the classpath. You can either use the `-classpath JVM` option in the `appclient` command or set the `CLASSPATH` environment variable. For more information about setting a classpath, see [Setting the Class Path, Oracle Solaris Version \(http://java.sun.com/javase/6/docs/technotes/tools/solaris/classpath.html\)](http://java.sun.com/javase/6/docs/technotes/tools/solaris/classpath.html) or [Setting the Class Path, Windows Version \(http://java.sun.com/javase/6/docs/technotes/tools/windows/classpath.html\)](http://java.sun.com/javase/6/docs/technotes/tools/windows/classpath.html).

application-options

optional; you can set client application arguments.

Examples EXAMPLE 1 Using the `appclient` command

```
appclient -splash welcome.jpg -xml sun-acc.xml -jar myclientapp.jar scott sample
```

Where: `welcome.jpg` is a splash screen specified by the `-splash JVM` option, `sun-acc.xml` is the name of the client configuration XML file, `myclientapp.jar` is the client application `.jar` file, and `scott` and `sample` are arguments to pass to the application. If `welcome.jpg`, `sun-acc.xml`, and `myclientapp.jar` are not in the current directory, you must give the absolute path locations; otherwise the relative paths are used. The relative path is relative to the directory where the command is being executed.

Attributes See `attributes(5)` for descriptions of the following attributes:

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Interface Stability	Unstable

See Also `get-client-stubs(1)`

`asadmin(1M)`, `package-appclient(1M)`

