

Setting Environment Variables in Descriptors

[Server descriptors](#) and [IceBox descriptors](#) may specify environment variables that the node will define when starting a server. An environment variable definition uses the familiar *name=value* syntax, and you can also refer to other environment variables within the value. The exact syntax for variable references depends on the platform on which the server's descriptor is deployed.

On a Unix platform, the Bourne shell syntax is required:

```
LD_LIBRARY_PATH=/opt/Ice/lib:$LD_LIBRARY_PATH
```

On a Windows platform, the syntax uses the conventional style:

```
PATH=C:\Ice\lib;%PATH%
```

In XML, the `env` element supplies a definition for an environment variable:

XML

```
<node name="UnixBox">
  <server id="UnixServer" exe="/opt/app/bin/server" ...>
    <env>LD_LIBRARY_PATH=/opt/Ice/lib:$LD_LIBRARY_PATH</env>
    ...
  </server>
</node>
<node name="WindowsBox">
  <server id="WindowsServer" exe="C:/app/bin/server.exe" ...>
    <env>PATH=C:\Ice\lib;%PATH%</env>
    ...
  </server>
</node>
```

If a value refers to an environment variable that is not defined, the reference is substituted with an empty string.

Environment variable definitions may also refer to [descriptor variables](#) and [template parameters](#):

XML

```
<node name="UnixBox">
  <server id="UnixServer" exe="/opt/app/bin/server" ...>
    <env>PATH=${server.distrib}/bin:$PATH</env>
    ...
  </server>
</node>
```

On Unix, an environment variable `VAR` can be referenced as `$VAR` or `${VAR}`. You must be careful when using the latter syntax because IceGrid assumes `${VAR}` refers to a descriptor variable or parameter and will report an error if no match is found. If you prefer to use this style to refer to environment variables, you must escape these occurrences as shown in the example below:

```
<node name="UnixBox">
  <server id="UnixServer" exe="/opt/app/bin/server" ...>
    <env>PATH=${server.distrib}/bin:${PATH}</env>
    ...
  </server>
</node>
```

IceGrid does not attempt to perform [substitution](#) on `$$PATH`, but rather removes the leading `$` character and then performs environment variable substitution on ``${PATH}`.

[See Also](#)

- [Server Descriptor Element](#)
- [IceBox Descriptor Element](#)
- [Using Descriptor Variables and Parameters](#)