

# Guide to Updating the isCOBOL Software Version in a Standalone, Fat Client or Thin Client Production Environment

---

A common topic to all three scenarios is the update of the license.

When your update is for a later year release (ex: 2021R# to 2022R#), then you need to request licenses for your products for that year and update the properties file containing your license keys.

To learn more about licenses, review the video on Veryant's YouTube channel:

<https://youtu.be/R6ScVpdT1XM>.

## 1) Updating the isCOBOL production environment in standalone mode

If you are installing a later build of the same release (ex: 2022R1\_1060.5 to 2022R1\_1060.8), the setup program will install in the same folder and that's it, you do not need further steps to continue working.

If you are installing a different release, it would install in a different folder by default, though an alternative case is to have a generic folder for the installation (without the release number).

*Example with the same folder name:*

You had the 2021R1 runtime in c:isCOBOL

You install the 2022R1 runtime in c:isCOBOL

*Example with different folder names (default):*

You had: C:VeryantisCOBOL\_SDK2021R1

And you update to: C:VeryantisCOBOL\_SDK2022R1

On Windows if you are installing in the same generic folder then you can install on top of the old version and there are no further steps.

On Linux if you want to install in the same generic folder, you will first install in a different folder name (usually the default) and then move the installed files and folders from there to the generic folder.

After that no further steps are needed.

If you install the new isCOBOL version in a different folder name (default) and want to keep that different folder name, then you have to update the scripts you use to start your application, so they point to the new installation.

For example: (example for 2022R1 and JDK 8u321):

*Windows*

---

```
set ISCOBOL=C:\Veryant\isCOBOL_SDK2022R1;set ISCOBOL_JRE_ROOT=C:\Program Files\Java\jre1.8.0_321;set CLASSPATH=%ISCOBOL%\lib*;%ISCOBOL%\jars*;%ISCOBOL_JRE_ROOT%\lib*;%CLASSPATH%;PATH=%ISCOBOL%\in;%ISCOBOL_JRE_ROOT%\in;%PATH%
```

## Linux

```
export ISCOBOL=/opt/veryant/isCOBOL_SDK2022R1;export ISCOBOL_JDK_ROOT=/opt/jre1.8.0_321;export CLASSPATH=$ISCOBOL/lib/*:$ISCOBOL/jars/*:$ISCOBOL_JRE_ROOT/lib/*:$CLASSPATH;export PATH=$ISCOBOL/bin:$ISCOBOL_JRE_ROOT/bin:$PATH
```

On Windows you have the option to set the system environment variables during setup rather than having to set them in each start-up script.

Just respond 'Yes' when the Installation Wizard asks that question during setup.

You can further edit those computer environment variables later to remove references to the prior version.

The last scenario for this case is if you copy/paste the isCOBOL folder structure from another computer to your standalone production computers instead of using the setup program.

- You can copy/paste to a generic folder name be done.
- Or you can copy/paste that structure to a new folder name and change scripts or update the computer environment variables manually as described above.

The usual folder structure to copy in this scenario is the /lib, /bin and /jars subfolders of the isCOBOL installation, though if you normally include other subfolders then continue doing so.

## 2) Updating the isCOBOL production environment in Fat Client mode

In this environment the isCOBOL File Server and data files are on a central server and the isCOBOL runtime with all the programs of your application are installed on every client.

If the server is Windows and you use the Windows Service for the isCOBOL File Server, first uninstall that service using this command:

```
isserver -uninstall
```

. Then use the setup program to install the new release and have it enable/install the Windows service for the file server on the same port as before.

If the server is Linux/Unix and you use the daemon for the isCOBOL File Server, stop the daemon with the command:

```
isserver stop
```

Then use the setup program to install the new release and restart with the command `isserver start`.

If you start the File Server with a script, stop/kill the file server first and then install the new

release in a new folder or the same folder as explained in section 1.

If you use the same installation folder then no further steps are needed before running the script to start the file server.

If you are using a new installation folder, update the script to point to the new installation folder before running that script.

For instance (example for 2022R1 and JDK 8u321):

### *Windows*

```
set ISCOBOL=C:\Veryant\isCOBOL_SDK2022R1;set ISCOBOL_JRE_ROOT=C:\Program Files\Java\jre1.8.0_321;set CLASSPATH=%ISCOBOL%\lib*;%ISCOBOL%\jars*;%ISCOBOL_JRE_ROOT%\lib*;%CLASSPATH%;PATH=%ISCOBOL%\bin;%ISCOBOL_JRE_ROOT%\bin;%PATH%
```

### *Linux*

```
export ISCOBOL=/opt/veryant/isCOBOL_SDK2022R1;export ISCOBOL_JDK_ROOT=/opt/jre1.8.0_321;export CLASSPATH=$ISCOBOL/lib/*:$ISCOBOL/jars/*:$ISCOBOL_JRE_ROOT/lib/*:$CLASSPATH;export PATH=$ISCOBOL/bin:$ISCOBOL_JRE_ROOT/bin:$PATH
```

Here again, if the server installation is on Windows and you use the computer environment variables, then you can have the setup program to update those variables for you (or you update them manually in the computer's environment variables if you do not use the setup wizard) and no further changes to any script is needed before restarting the file server.

To update the client's runtime and the client's application programs you can use the `isupdater` utility, which is explained in more detail in Veryant's documentation.

### **3) Updating the isCOBOL production environment in Thin Client mode**

In this scenario, you have the isCOBOL Application Server, the application programs, and data files on a central server or servers and the clients only have the thin-client runtime installed.

If the server is Windows and you use the Windows Service for the Application server, first uninstall that windows service using this command:

```
isserver -uninstall
```

Then use the setup program to install the new release and have it enable/install the Windows service for the Application server on the same port as before.

If the server is Linux/Unix and you use the daemon for the Application server, stop the daemon with the command:

```
isserver stop
```

Then use the setup program to install the new release and restart with the command:

```
isserver start
```

If you start the Application server with a script, then stop/kill the Application server first and install the new release in a new folder or the same folder as explained in section 1 above.

If you use the same installation folder then no further steps are needed before running the script to start the Application server.

If you are using a new installation folder, update the script to point to the new installation folder before running that script. For example: (example for 2022R1 and JDK 8u321):

### *Windows*

```
set ISCOBOL=C:\Veryant\isCOBOL_SDK2022R1;set ISCOBOL_JRE_ROOT=C:\Program Files\Java\jre1.8.0_321;set CLASSPATH=%ISCOBOL%\lib*;%ISCOBOL%\jars*;%ISCOBOL_JRE_ROOT%\lib*;%CLASSPATH;%PATH=%ISCOBOL%\in;%ISCOBOL_JRE_ROOT%\in;%PATH%
```

### *Linux*

```
export ISCOBOL=/opt/veryant/isCOBOL_SDK2022R1;export ISCOBOL_JDK_ROOT=/opt/jre1.8.0_321;export CLASSPATH=$ISCOBOL/lib/*:$ISCOBOL/jars/*:$ISCOBOL_JRE_ROOT/lib/*:$CLASSPATH;export PATH=$ISCOBOL/bin:$ISCOBOL_JRE_ROOT/bin:$PATH
```

Again, if the server installation is on Windows and you use the computer environment variables, then you can have the setup program to update those variables for you (or you update them manually in the computer's environment variables if you do not use the setup wizard) and no further changes to any script is needed before restarting the Application server.

The application programs on the central server can be updated (if they were changed and/or compiled with the new release as well) by just FTPing the new application programs to the classes folder used on your central server.

To update the Thin-client runtime on every client, you should use the isupdater utility, which is explained in more detail in Veryant's documentation as well as in this video on Veryant's YouTube channel: <https://youtu.be/ACzsaEeY1c>.

Online URL: <https://support.veryant.com/support/phpkb/article.php?id=330>