
Article

[Pravin Barton](#) · Apr 26, 2017 1m read

Moving a global to a new database using extended global reference

Say you have a global in one database that you instead want to map from a different database. If you just create a global mapping to the new database, the existing global will become inaccessible because it still lives on the old database. The documentation notes this problem [here](#) but doesn't give details on how to fix it.

When you create the global mapping you need to explicitly move the existing global to the new database. This is possible using the merge command along with [extended global reference](#). The environment syntax lets you reference a global in a specific namespace, directory, or remote system. For example, I can reference the ^pbarton global in the USER database with the following syntax:

```
zw ^|^c:/cache/user/|pbarton
```

if the USER database lives in the directory "C:/cache/user/".

In order to move this global to the USER-CODE database that lives in the directory "C:/cache/usercode/", I can use the merge command while explicitly specifying the environment in my global references:

```
merge ^|^c:/cache/usercode/|pbarton = ^|^c:/cache/user/|pbarton
```

[#Caché #ObjectScript #Mapping](#)

Source

URL:<https://community.intersystems.com/post/moving-global-new-database-using-extended-global-reference>