

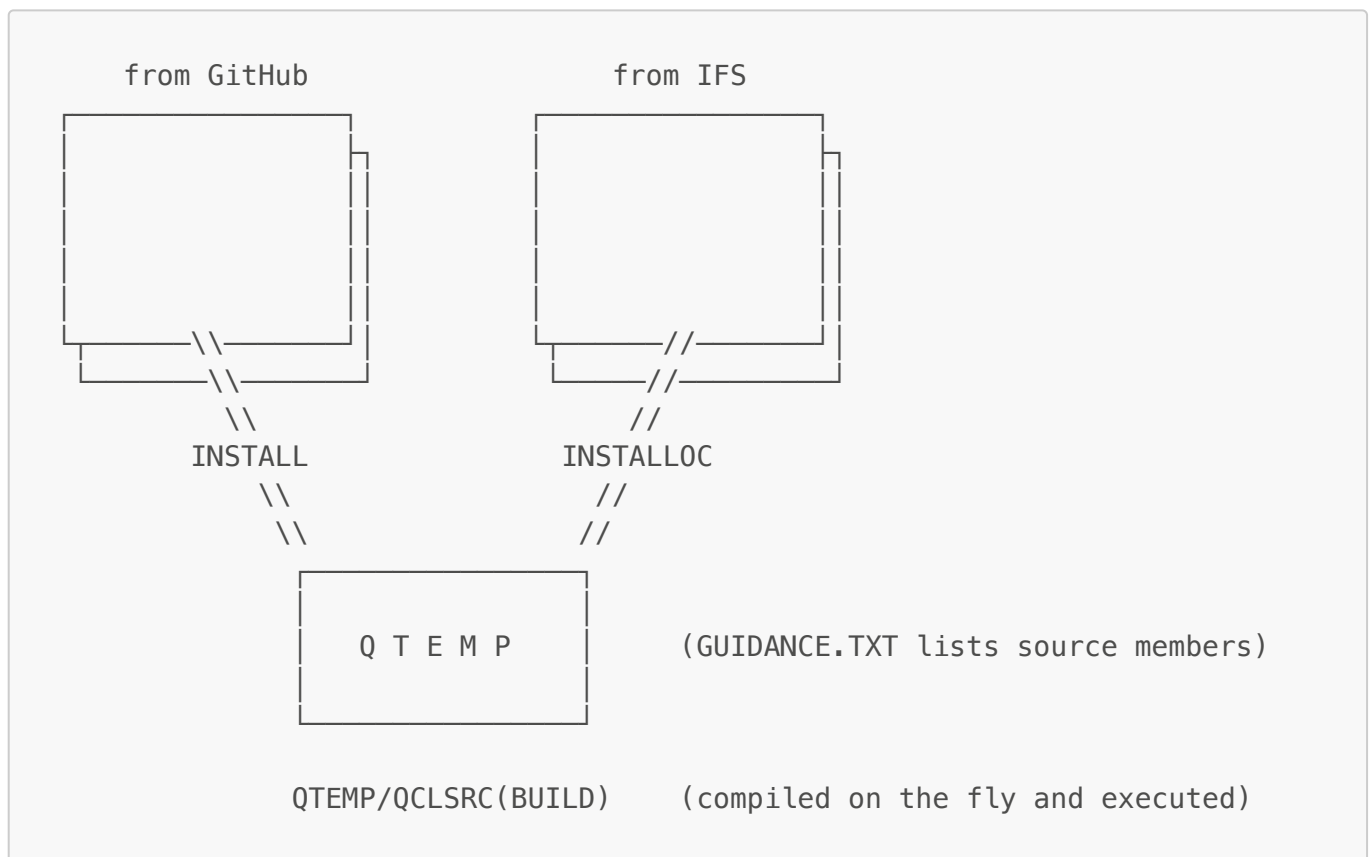
PASERIE (V1R0M9)

Utility for Source-Level Distribution in IBM i

This utility helps you maintain your IBM i source code on GitHub. Once available on a target system, the utility allows you to install your packages in a very simple way, directly accessing your repositories on GitHub or the repositories of other developers that authorized you.

The suite provides a total of three commands:

- For end users of a package:
 - **PASERIE/INSTALL** the main command focusing on the installation phase of an already developed and GitHub-released package.
- For developers:
 - **PASERIE/INSTALLOD** the command used in the development phase to test the installation of a package (planned to be released on GitHub or not) directly from a directory in the IFS.
 - **PASERIE/LIBCLONE** the command used to jump-start the packaging (consistently with PASERIE tools' conventions) from an existing native library.



INSTALLATION

Let's first create an empty SAVEFILE named **QGPL/PASERIE**:

```
CRTSAVF FILE(QGPL/PASERIE) TEXT('Paserie Save File')
```

Then, from **CALL QP2TERM**, we verify that **curl** is installed:

```
$
> PATH=/QOpenSys/pkgsrc/bin:$PATH
$
> export PATH
$
> which curl
/QOpenSys/pkgsrc/bin/curl
```

Using curl, we download 2 archives:

```
curl -o PaseOssFloating.tar
      https://www.andrearibuoli.it/paserie/PaseOssFloating.tar
```

```
curl -o /QSYS.LIB/QGPL.LIB/PASERIE.FILE
      https://www.andrearibuoli.it/paserie/PASERIE.SAVF
```

From the home path of the user that will be using **PASERIE/INSTALL** we install **PaseOss** folder by untarring *PaseOssFloating* tar file:

```
cd ~
tar xvf PaseOssFloating.tar
```

Now, after exiting from the QP2TERM session, we can perform the **RSTLIB** for PASERIE:

```
RSTLIB SAVLIB(PASERIE) DEV(*SAVF) SAVF(QGPL/PASERIE)
```

PARAMETERS FOR THE THREE COMMANDS WITH SCREEN SHOTS

KWD	INSTALL	INSTALLOD	LIBCLONE
REPO_OWNER	yes		
REPOSITORY	yes		
YOURGITPAT	yes		
SRCLIB			yes
LOCALPATH		yes	yes
TGTLIB	yes	yes	

KWD	INSTALL	INSTALLOC	LIBCLONE
TGTRLS	yes	yes	
DEVOPT	yes	yes	yes
LOGOUTPUT	yes	yes	yes
VERBOSE	yes	yes	yes

```

PASERIE Installer          V1R0M2 (INSTALL)

Type choices, press Enter.

GitHub repository owner . . . . REPO_OWNER
Repository . . . . . REPOSITORY
GitHub personal access token . . YOURGITPAT

Target library . . . . . TGTLIB      *REPOSITORY
Target release . . . . . TGTRLS     *CURRENT

Additional Parameters

Developemnt option . . . . . DEVOPT   N
Job lob output . . . . . LOGOUTPUT  *PND
Verbose . . . . . VERBOSE          N

Bottom
```

```

PASERIE Local Installer V1R0M2 (INSTALLOC)

Type choices, press Enter.

Project directory . . . . .

Target library . . . . . *LOCAL_PATH Character value, *LOCAL_PATH
Target release . . . . . *CURRENT   Character value, *CURRENT...

Additional Parameters

Developemnt option . . . . . N      Character value
Job lob output . . . . . *PND      Character value, *SAME...
Verbose . . . . . N              Character value

Bottom
```

```

PASERIE Library Clone  V1R0M2 (LIBCLONE)

Type choices, press Enter.

Source library . . . . . SRCLIB
Target project directory . . . . LOCAL_PATH  *SRCLIB

Additional Parameters

Developemnt option . . . . . DEVOPT      N
Job lob output . . . . . LOGOUTPUT    *PND
Verbose . . . . . VERBOSE      N

Bottom
```

THE NAME

pastry ==> *pâ-tis-se-rie* ==> **PASERIE**

HINTS FOR DEVELOPERS

The minimum provision to enable **PASERIE/INSTALL** handling your "native" IBM i GitHub repository is:

- creating a file named **GUIDANCE.TXT** in the root of the repository and
- creating a directory named **QCLSRC** and a **QCLSRC/BUILD.CLLE** ILE CL source

The installer will use the token passed with **YOURGITPDA** (*GitHub personal access token*) to access GitHub APIs and download **GUIDANCE.TXT** in memory. The content is a list of member files that will be created in the **QTEMP** of the batch job. Once all files are transfered the job will attempt the compilation of QTEMP/QCLSRC(BUILD) member file into QTEMP/BUILD *PGM. If successful, it will pass control to your build procedure.

The expected input parameters of a well-behaving **QCLSRC/BUILD.CLLE** are the following (in this order):

```
DCL VAR(&DEVOPT) TYPE(*CHAR) LEN(1)
DCL VAR(&TGTRLS) TYPE(*CHAR) LEN(10)
DCL VAR(&TGTLIB) TYPE(*CHAR) LEN(10)
```

that default to 'N', '*CURRENT', and '*PACKAGEN' if not set in **PASERIE/INSTALL**, or **PASERIE/INSTALLLOC**, corresponding parameters.

The **PASERIE/LIBCLONE** utility generates a directory with all the required PASERIE-related objects. It also introduces a dependency on the **TMKMAKE** utility (from **QUSRT00L** library). I have re-packaged it for a plain installation with **PASERIE/INSTALL** (please contact me if you need help in installing it: andrea.ribuoli@yahoo.com)

HANDS ON

Let us take confidence with **PASERIE/LIBCLONE**.

First we create a new native library named **SIMPLE**:

CRTL **LIB(SIMPLE)**

Now we create a source file named **QCLSRC** with a member named **GREETINGS**:

```
CRTSRCPF FILE(SIMPLE/QCLSRC)
        MBR(GREETINGS)
        TEXT('Help me understand PASERIE tools')
```

Let us assign the **CLLE** SEU-type to our member file:

CHGPFM **FILE(SIMPLE/QCLSRC)** **MBR(GREETINGS)** **SRCTYPE(CLLE)**

Let us edit the source file:

EDTF **FILE(SIMPLE/QCLSRC)** **MBR(GREETINGS)**

and enter a simple CL source like:

```
Edit File: SIMPLE/QCLSRC(GREETINGS)
Record :      1    of      3 by 10
Control : _____

CMD ..+....2....+....3....+....4....+....5....+....6....
      *****Beginning of data*****
      PGM
      SNDPGMMSG MSG('Your first use of PASERIE/LIBCLONE')
      ENDPGM
      *****End of Data*****
```

Now we issue the following command:

PASERIE/LIBCLONE **SRCLIB(SIMPLE)**

We should receive:

```
Selection or command
===>
```

```
F3=Exit   F4=Prompt
F23=Set initial menu
LIBCLONE_B completed.
```

Now let us run `WRKLNK OBJ(SIMPLE)` and use option `5` to enter the directory. We should find:

Opt	Object link	Type	Attribute	Text
	GUIDANCE.TXT	STMF		
	QCLSRC	DIR		
	QMAKSRC	DIR		

The content of `GUIDANCE.TXT` will be:

```
*****Beginning of data*****
QCLSRC      GREETINGS      Help me understand PASERIE tools
QCLSRC      BUILD      CLLE      BUILD CLLE auto-generated
QMAKSRC     BUILD      TXT      BUILD Makefile auto-generated
*****End of Data*****
```

Inside `QCLSRC` directory you will find two source members:

Opt	Object link	Type
	BUILD.CLLE	STMF
	GREETINGS.CLLE	STMF

And there is also a `QMAKSRC` with a `BUILD` member.

Assuming you have a `TMKMAKE` library with a working `TMKMAKE` command you will be able to run successfully:

`PASERIE/INSTALLLOC LOCAL_PATH(SIMPLE)`

Now issuing `CALL PGM(SIMPLE/GREETINGS)` we will get:

```
Selection or command
===>

F3=Exit   F4=Prompt   F9=Retrieve
F23=Set initial menu
Your first use of PASERIE/LIBCLONE
```