



BSF4ooRexx: From 641 GA *Update* to 850 Beta



BSF4ooRexx 641: Java 6 and ooRexx 4.1 or later
BSF4ooRexx850: Java 8 and ooRexx 5.0 or later

International RexxLA Symposium, 2022-09 (<https://www.RexxLA.org>)



- Imminent BSF4ooRexx 641 GA update
 - Work, changes and new features since the GA release on January 31st, 2022
 - Attention! Removed `jsr223` fragment from package name
 - Planned release
 - Maintenance status
- Work on BSF4ooRexx 850
 - Based on 641 GA update
 - Work and changes
 - New features

BSF4ooRexx 641 GA Update, Changes 1



- 641
 - `641 % 100: 6` ... baseline Java 6 (a.k.a. 1. 6), newer o.k.
 - `641 // 100: 41` ... baseline ooRexx 4.1, newer o.k.
- Fixed memory leak in 7/24 deployments
 - Changed finalization of Java objects
 - Reworked class, added new enum value `REXX_SCRIPT_ENGINE`
 - `org.rexxla.bsf.engines.rexx.RexxCleanupRef`
 - Can be used to gain insights about the number of created `RexxProxy`, `RexxEngine` and `RexxScriptEngine` objects, and how many of these have been finalized so far
 - Developed new class to make it easier to analyze the Java registry
 - New class that allows to analyze the current `RexxProxy` objects in the Java registry
 - `org.rexxla.bsf.engines.rexx.RexxAnalyzeRegistry`
 - E.g. method `public static String getAnalyzedDataAsString()`
 - However note: this class depends on Java 8 and can only be used using Java 8 or later!



BSF4ooRexx 641 GA Update, Changes 2



- Fixed memory leak in 7/24 deployments (continued)
 - Reworked class, added instance counter to ease debugging
 - `org.rexxla.bsf.engines.rexx.RexxEngine`
- Attention: removed `jsr223` fragment from package name
 - From `org.rexxla.bsf.engines.rexx.jsr223`
 - To `org.rexxla.bsf.engines.rexx`
 - Affected classes
 - `RexxScriptEngine`
 - `RexxScriptEngineFactory`
 - `RexxCompiledScript`
 - Adjusted all samples that would explicitly use one of these three classes



BSF4ooRexx 641 GA Update, Changes 3



- New scripts to run ooRexx via Java on the command line
 - `rexxj3.cmd` (Windows), `rexxj3.sh` (Unix)
 - Windows: `rexxj3 someRexxProgram.rex`
 - Unix: `rexxj.sh someRexxProgram.rex`
 - Honor environment variable `JAVA_HOME`
 - If present uses the Java version pointed to by `JAVA_HOME`
- New logic to demand load Java via ooRexx
 - Honors environment variable `JAVA_HOME` first
 - If present uses the Java version pointed to by `JAVA_HOME`, e.g. Windows
 - `%JAVA_HOME%\bin\server\jvm.dll` first, then
 - `%JAVA_HOME%\jre\bin\server\jvm.dll`

BSF4ooRexx 641 GA Update, Changes 4



- Change (Windows only)
 - The external Rexx function `BsfVersion()` now includes the string `DEBUG`
 - Makes it easy to distinguish a debug from a release build at runtime
- New subfunction for external Rexx function `BSF()`
 - `BSF("simpleExecute" [, rexxCode [, stringArgument]])`
 - This function will create a new Rexx interpreter instance (RII)
 - If `rexxCode` is supplied that Rexx program runs on it, optionally returning a return value
 - If `stringArgument` is supplied it will be used as the single argument for `rexxCode`
 - The function will terminate the newly created RII before returning any result
 - Can be used for
 - Testing Rexx interpreter instances for debugging
 - Running Rexx programs insulated in a separate, temporary Rexx interpreter instance



BSF4ooRexx 641 GA Update, Changes 5



- New feature: create portable installation zip archives
 - Unpack the BSF4ooRexx installation zip-archive,
 - A subdirectory `bsf4oorexx` gets created
 - Run the script `bsf4oorexx\install\portable\createPortableB4R.rex`
 - Supply as blank delimited arguments
 - `op_sys ... L(inux) or W(indows) or D(arwin, i.e. MacOS)`
 - `bitness ... 32 or 64`
 - `machine ... "x" (Intel) amd (Intel/AMD) arm (ARM) s390x (IBM mainframe Linux)`
 - If no arguments supplied the script will use ooRexx to determine the operating system, the bitness and the machine type
 - There will be two portable zip archives created
 - Binary only, e.g. `bsf4oorexx_v850.20220904-Windows-amd64-portable-UNO-runtime.zip`
 - Full package, e.g. `bsf4oorexx_v850.20220904-Windows-amd64-portable-UNO.zip`



BSF4ooRexx 641 GA Update, Changes 6



- New feature: create portable installation zip archives (continued)
 - Unpack either, runtime or full zip archive
 - Change into the unzipped directory and issue
`rex x setupbsf4oorex x .rex`
 - This script will create the following tailored scripts for e.g. Windows
 - `rex xj .cmd` ... a tailored script to run ooRexx programs that use Java, e.g.
`rex xj testbsf4oorex x .rex` ... gives version information about Java, ooRexx and BSF4ooRexx
`rex xj testuno .rex` ... lists the OpenOffice or LibreOffice path settings, if installed
 - `setenv2bsf4oorex x .cmd` ... a script that sets the environment
 - If having the unzipped portable version on an USB device and you switch PCs
 - Merely rerun
`rex x setupbsf4oorex x .rex`
which will recreate the local sensitive scripts `rex xj .cmd` and `setenv2bsf4oorex x .cmd`

BSF4ooRexx 641 GA Update, Changes 7



- Samples

- [samples/JavaFX/fxml_20/demoListView.rex](#)
[samples/JavaFX/fxml_25/demoTableViewSimple.rex](#)
[samples/JavaFX/fxml_26/demoTableViewComboBoxCell.rex](#)
[samples/JavaFX/fxml_27/demoTableView.rex](#) ... changes: do not start with debug output on anymore, corrected a few subtle bugs in the user interface by using BSF4ooRexx' public routine `box.strictArg()` instead of `box()`
- [samples/4-020_EuroCalcJava.rxj](#) ... added Croatian Kuna fixed rate of exchange (Croatia will introduce the Euro on January 1st, 2023)

BSF4ooRexx 641 GA Update, Changes 8



- Samples
 - [samples/4-070_BezierCurves.rxx](#) ... new sample that demonstrates an ooRexx solution for *Ruurd Idenburg's* NetRexx demo

Exploring Bezier curves with:

ooRexx

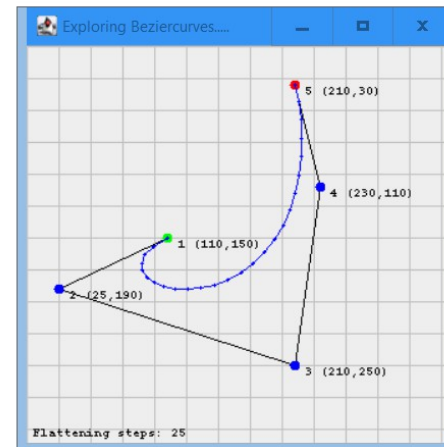
ooRexx is a project of the Rexx Language Association ([RexxLA](#)). This solution uses the ooRexx-Java bridge [BSF4ooRexx](#) which is a project of the Rexx Language Association ([RexxLA](#)) as well.

The code is based on the information provided on [this website](#) (<https://pomac.github.io/bezierinfo/>).

You can use this code in the following way:

- Execution starts with a predefined set of startpoint, endpoint and 3 controlpoints.
- With the left mousebutton all points can be dragged.
- Clicking the left mousebutton on one of the points will add an additional controlpoint between the one clicked and the next point.
- Clicking the right mousebutton on one of the points will remove that controlpoint.
- Double clicking the left mousebutton allows you the change the number of flattening steps.

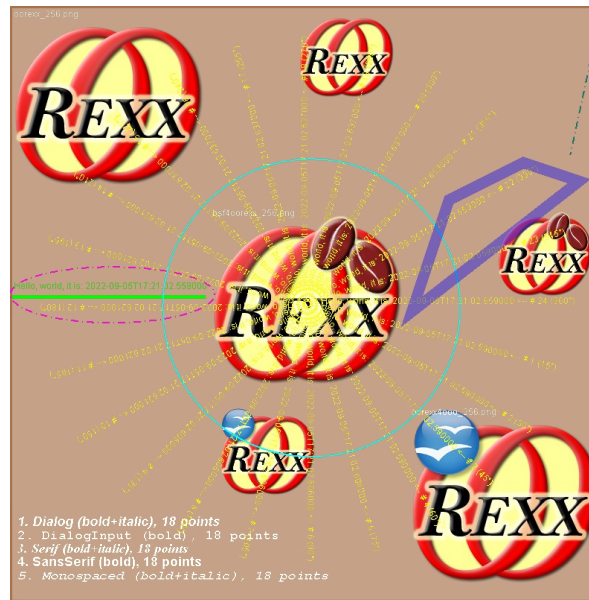
This ooRexx solution is based on Ruurd Idenburg's NetRexx solution, kudos to Ruurd who wrote: *Go ahead, have some fun, I hope! Ruurd Idenburg*



BSF4ooRexx 641 GA Update, Changes 9



- Samples
 - [samples/3-100_create_bitmap.rxj](#) ... new sample that demonstrates how to create a bitmap (an image) with Java's *Graphics2D*, the resulting bitmap gets saved in the user's home directory under the name [3-100_create_bitmap.png](#) and is shown using the appropriate operating system's command



BSF4ooRexx 641 GA Update, Changes 10



- Samples (continued)
 - [samples/1-062_DateRgfCatholicEaster.rxj](#) ... new sample taking advantage of BSF4ooRexx' `org.oorexx.datergf.DateRGF` class to calculate the catholic Easter related holidays

```
Year/Jahr: [2022] -----
Palm Sunday/Palmsonntag           [2022-04-10] (Easter Sunday - 7 days)
Good Friday/Karfreitag:           [2022-04-15] (Easter Sunday - 2 days)
Easter Sunday/Ostersonntag:       [2022-04-17]
Ascension Day/Christi Himmelfahrt: [2022-05-26] (Easter Sunday + 39 days)
Pentecost/Pfingstsonntag:         [2022-06-05] (Easter Sunday + 49 days)
Whit Monday/Pfingstmontag:        [2022-06-06] (Easter Sunday + 50 days)
Corpus Christi/Fronleichnam:      [2022-06-16] (Easter Sunday + 60 days)

Year/Jahr: [2023] -----
Palm Sunday/Palmsonntag           [2023-04-02] (Easter Sunday - 7 days)
Good Friday/Karfreitag:           [2023-04-07] (Easter Sunday - 2 days)
Easter Sunday/Ostersonntag:       [2023-04-09]
Ascension Day/Christi Himmelfahrt: [2023-05-18] (Easter Sunday + 39 days)
Pentecost/Pfingstsonntag:         [2023-05-28] (Easter Sunday + 49 days)
Whit Monday/Pfingstmontag:        [2023-05-29] (Easter Sunday + 50 days)
Corpus Christi/Fronleichnam:      [2023-06-08] (Easter Sunday + 60 days)

Year/Jahr: [2024] -----
Palm Sunday/Palmsonntag           [2024-03-24] (Easter Sunday - 7 days)
Good Friday/Karfreitag:           [2024-03-29] (Easter Sunday - 2 days)
Easter Sunday/Ostersonntag:       [2024-03-31]
```

BSF4ooRexx 641 GA Update, Changes 11



- Samples (continued)
 - [samples/1-064_DateRgfOrthodoxEaster.rxx](#) ... new sample taking advantage of BSF4ooRexx' [org.oorexx.datergf.DateRGF](#) class to calculate the orthodox Easter related holidays

Year/Jahr: [2022] (Dates are given in Gregorian calendar)-----

Lazarus Saturday/Georgstag	[2022-04-16]	(Easter Sunday - 8 days)
Palm Sunday/Palmsonntag	[2022-04-17]	(Easter Sunday - 7 days)
Holy Friday/Karfreitag:	[2022-04-22]	(Easter Sunday - 2 days)
Easter Sunday/Ostersonntag:	[2022-04-24]	
Ascension Day/Christi Himmelfahrt:	[2022-06-02]	(Easter Sunday + 39 days)
Pentecost/Pfingstsonntag:	[2022-06-12]	(Easter Sunday + 49 days)
Whit Monday/Pfingstmontag:	[2022-06-13]	(Easter Sunday + 50 days)

Year/Jahr: [2023] (Dates are given in Gregorian calendar)-----

Lazarus Saturday/Georgstag	[2023-04-08]	(Easter Sunday - 8 days)
Palm Sunday/Palmsonntag	[2023-04-09]	(Easter Sunday - 7 days)
Holy Friday/Karfreitag:	[2023-04-14]	(Easter Sunday - 2 days)
Easter Sunday/Ostersonntag:	[2023-04-16]	
Ascension Day/Christi Himmelfahrt:	[2023-05-25]	(Easter Sunday + 39 days)
Pentecost/Pfingstsonntag:	[2023-06-04]	(Easter Sunday + 49 days)
Whit Monday/Pfingstmontag:	[2023-06-05]	(Easter Sunday + 50 days)

Year/Jahr: [2024] (Dates are given in Gregorian calendar)-----

Lazarus Saturday/Georgstag	[2024-04-27]	(Easter Sunday - 8 days)
Palm Sunday/Palmsonntag	[2024-04-28]	(Easter Sunday - 7 days)

BSF4ooRexx 641 GA Update, Changes 12



- Samples (continued)
 - [samples/JavaFX/fxml_03/fxml/registration.fxml](#) ... added third gender '*neuter*'
 - [samples/JavaFX/fxml_05](#) ... updated the *ControlsFX* jar files for Java 8 and Java 11 (removed the outdated Java 9 version)
 - added [staff.rex](#), [run_on_java_8.cmd](#), [run_on_java_8.sh](#), [run_on_java_11.cmd](#), [run_on_java_11.sh](#)
 - demonstrate how to set `CLASSPATH` depending on the Java version in use before running [staff.rxj](#) which depends on the correct *ControlsFX* jar



BSF4ooRexx850 Beta, 1



- Based on 641 GA update
- Created svn [branches/850](#) on August 10th 2022
- Baseline Java 8
 - E.g. allows for defining [default](#) methods in interface classes
- Baseline ooRexx 5.0
 - Allows for exploiting new APIs (cf. [rexxapi.pdf](#)) like [SendMessageScoped\(\)](#) but also newly introduced features like redirecting command handlers via the API [AddCommandEnvironment\(\)](#) and related new APIs



BSF4ooRexx850 Beta, 2



- Java 8 related changes
 - Refactoring the exit and command handler interface classes
 - Change from abstract to interface classes
 - Changing the static methods to default methods in the new interface classes
 - Creating a handler related subtree with
 - `RexxHandler` (handler's root interface class)
 - `RexxExitHandler` (interface class inheriting from `RexxHandler`)
 - `RexxCommandHandler` (interface class inheriting from `RexxHandler`)
 - `RexxRedirectingHandler` (**new class**, inheriting from `RexxCommandHandler`)
 - Allow handlers to be implemented in ooRexx
 - `AbstractExitHandler` ... `RexxExitHandler`'s *handleExit()*
 - `AbstractDirectCommandHandler` ... `RexxCommandHandler`'s *handleCommand()*
 - `AbstractRedirectingCommandHandler` ... `RexxCommandHandler`'s *handleCommand()*



- ooRexx 5.0 related changes
 - [RexxProxy](#) Java class extended with new method [sendMessageScoped\(...\)](#)
 - Allows to override the normal lookup of methods by supplying a superclass ("scope") where the search should begin for the method according to the message name
 - RexxConfiguration
 - Allows configuring a Rexx interpreter instance before first usage to e.g.
 - Define exit and command handlers, default address environment to start out with, additional Rexx file extensions, preloaded libraries (cf. rexxapi.pdf)
 - Now the new redirecting command handlers are supported!
 - New external BSF function named [BsfCommandHandler\(\)](#)
 - call [BsfCommandHandler 'add', environmentName, javaCommandHandler](#)
 - [BsfCommandHandler\('list'\)](#) ... returns an array of currently defined command handlers





- ooRexx 5.0 related changes (continued)
 - New external BSF function named `BsfCommandHandler()`
 - Add at runtime (!) a new direct or redirecting command handler implemented in Java
call `BsfCommandHandler 'add', environmentName, javaCommandHandler`
 - Return a list of currently defined Java command handlers together with the environment names they serve, indicating whether the handlers are direct or redirecting
`BsfCommandHandler('list')` ... returns an array of currently defined command handlers
 - Note: Java command handlers can also be implemented in ooRexx taking advantage of BSF4ooRexx(850)!
 - Examples can be found in the subdirectories named
 - `samples\Java\handlers\commandHandlers\30_java_starter850/rexxonly`
 - `samples\Java\handlers\exitHandlers\04_RXMSQ/rexxonly`





- Added samples and unit tests
 - Exploring and demonstrating the newly added features
 - [samples/Java/handlers/commandHandlers/02_commandHandler850](#)
 - [samples/Java/handlers/commandHandlers/11_commandHandlers850](#)
 - [samples/Java/handlers/commandHandlers/20_redirectingCommandHandler850](#)
 - [samples/Java/handlers/commandHandlers/30_java_starter850](#)
 - [samples/Java/handlers/commandHandlers/30_java_starter850/rexxonly](#) (ooRexx only)
 - Same examples written in NetRexx
 - [samples/NetRexx/handlers/commandHandlers/02_commandHandler850](#)
 - [samples/NetRexx/handlers/commandHandlers/11_commandHandlers850](#)
 - [samples/NetRexx/handlers/commandHandlers/20_redirectingCommandHandler850](#)
 - [samples/NetRexx/handlers/commandHandlers/30_java_starter850](#)
 - [samples/NetRexx/handlers/commandHandlers/30_java_starter850/rexxonly](#) (ooRexx only)
- Unit tests to test the most important new additions



BSF4ooRexx850 Beta, 6



- Installation related changes
 - [BSF4ooRexx](#) installation directory gets renamed to [BSF4ooRexx850](#)
 - Allows to keep it separate from a BSF4ooRexx 641 directory
 - Dynamic link libraries, shared libraries renamed to
 - [BSF4ooRexx850.dll](#) (Windows)
 - [libBSF4ooRexx850.dylib](#) (MacOS)
 - [libBSF4ooRexx850.so](#) (Linux)
 - Allows to keep them separate and distinguishible from the BSF4ooRexx 641 libraries



BSF4ooRexx850 Beta, 7



- Current status
 - As stable as BSF4ooRexx 641
 - Open
 - Testing of installing and uninstalling the package
 - Change of directory and native library names may cause problems
 - Compiling the native library for all platforms
 - Once done a beta release will be made available from BSF4ooRexx' Sourceforge home
 - Seeking for shops/people to help compiling for the s390x platform
 - Future planned work
 - Redesign the installation logic
 - Change installation of Apache OpenOffice/LibreOffice (do not do it automatically)
 - Add *module-info.class* to allow BSF4ooRexx to be used as a proper module
 - Prerequisite: unbundle contained jar files
 - Planned to place all jar files into a *lib* subdirectory to allow referring with a wildcard to all

BSF4ooRexx: From 641 GA *Update* to 850 Beta



- Roundup
 - [BSF4ooRexx 641 GA update](#) pending
 - Currently missing native compilation for s390x architecture
 - Planned brief beta phase thereafter
 - No open, no known bugs
 - [BSF4ooRexx850 beta](#)
 - Based on 641 GA update
 - Development is finished
 - Once all native libraries have been created making it downloadable from Sourceforge
 - Quality is of type release: no open, no known bugs
- Questions ?