"Running Rexx from a USB Drive"

The 2020 International Rexx Symposium
Online ("Covid-19")
September 29th – October 1st 2020

© 2020 Rony G. Flatscher (Rony.Flatscher@wu.ac.at)
Wirtschaftsuniversität Wien, Austria (http://www.wu.ac.at)
Agenda

• ooRexx before 5.0
• The world with ooRexx 5.0 :-)
• Howto create "stick versions"
• Howto create and use "USB version"
• Demonstration
• Roundup
• Running "rexx" causes "rxapi" daemon to run
  – "rxapi"
    • Systemwide daemon, background process
    • Clashes with other versions of Rexx
      – Not possible to run ooRexx and Regina
        » Regina comes with a "regina" executable to not clash
      – Not possible to run 32- and 64-bit in parallel
  – Possible to run multiple Rexx interpreters
    • Sequentially
    • Need to stop (kill) the rxapi process
    • Need some technical insight
ooRexx before 5.0, 2

• Installing ooRexx 4.x
  – Systemwide
  – Administrative ("root") rights necessary!
  – A usual consequence
    • Not possible to get ooRexx installed on PCs in an organization, in a business
      – Users usually do not possess administrative rights
      – System administrators usually balk
        » "security concerns"
        » Software deployment policy
        » ...

• Locating the ooRexx libraries
  – Relative to the location of the binary
    • Unix (Linux "so", MacOSX "dylib")
      ../lib
    • Windows: same directory as binary

• The "rxapi" service has been revisited
  – Depends on the Rexx version
    • e.g., 5.0.0, 5.0.1, 5.1, etc.
    • Possible to have multiple ooRexx versions in parallel!
    • Possible to have 32- and 64-bit versions in parallel!
• Define a common directory structure

    oorexx/
    
    Darwin/
      x86_64/ ... bin/ ... include/... lib/ ... share/

    doc/

    Linux/
      x86/ ... bin/ ... include/... lib/ ... share/
      x86_64/ ... bin/ ... include/... lib/ ... share/

    Windows/
      x86/ ... bin/ ... include/... samples/
      x86_64/ ... bin/ ... include/... samples/
How to Create the "Stick" Versions, 2

- Collect the files after ooRexx got created
- Unix (Linux, MacOS)
  - Use the script "stickCreateUnix.sh"
    - Will create a "stick" zip-archive of the ooRexx interpreter with the needed structure
      - Directory named after the operating system ("uname -s")
      - Subdirectory named after the machine kind ("uname -m")
      - Subdirectory name bin/, include/, lib/, share/
  - Upon completion will copy the "stick" zip archive to the directory the script "stickCreateUnix.sh" resides in
    - Use that "stick" zip archive to create the USB stick version
      - Change into "oorexx" and unzip all created "stick" zip archives there
How to Create the "Stick" Versions, 3

• Collect the files after ooRexx got created
  – MacOS: after "make install"
    • Change into installation directory that contains the directories "bin", "include", "lib", "share"
    • Run the script "stickCreateUnix.sh"
  – Linux: after "cpack ./
    • locate and change into the subdirectory that contains the directories "bin", "include", "lib", "share"
      – You may want to use something like "find . -name rextry.rex" to locate it
    • Run the script "stickCreateUnix.sh"
> How to Create the "Stick" Versions, 4

- Collect the files after ooRexx got created
  - First run "nmake nsis_template_installer"
  - Then run the script "stickCreateCreateWindows.cmd"
    - "NSIS\files\Core\*" → "tmpStick\ooRexx\Windows\x86\_64\bin"  
    - "NSIS\files\DevLib\api\*" → "tmpStick\ooRexx\Windows\x86\_64\include"  
    - "NSIS\files\Samples\samples\*" → "tmpStick\ooRexx\Windows\x86\_64\samples"  
    - "NSIS\files\Docs\doc\*" → "tmpStick\ooRexx\doc"
Get Access to ooRexx on the USB Stick

• What you get
  – All five ooRexx versions on a single USB stick!
  – E.g., visiting friends, plugging in the USB stick and run
    a simple script off the "oorexx" directory and then run ooRexx off the stick, no matter from where
  – E.g., employee without administrative rights
    • Copy the USB stick content locally to the computer
    • Run the same simple script off the "oorexx" directory and from then on run ooRexx off the computer from everywhere
USB-Stick Scripts, 1

• Unix (Linux, MacOS)
  – Run "setExecutable.sh"
  – Run "create_Unix_scripts.sh"

    • Creates the following scripts in $HOME
      run_ooRexx64.sh, run_ooRexx32.sh (Linux)
      goto_ooRexx64.sh, goto_ooRexx32.sh (Linux)
      setup_environment4ooRexx64.sh, setup_environment4ooRexx32.sh (Linux)

  – Gotchas
    • Executable bit → "setExecutable.sh"

  – Invoke, e.g.,
    ~/run_ooRexx64.sh ... args or ~/run_ooRexx32.sh ... args
    ~/goto_ooRexx64.sh or ~/goto_ooRexx32.sh
    source ~/setup_environment4ooRexx64.sh or source ~/setup_environment4ooRexx6432.sh
• Windows
  – Run "create_Windows_scripts.cmd"
    • Creates the following scripts in %USERPROFILE%
      run_ooRexx64.cmd, run_ooRexx32.cmd
      goto_ooRexx64.sh, goto_ooRexx32.cmd
      setup_environment4ooRexx64.cmd, setup_environment4ooRexx32.cmd
    – Invoke, e.g.,
      %userprofile%\run_ooRexx64 ... args ...
      %userprofile%\run_ooRexx32 ... args ...
      %userprofile%\goto_ooRexx64
      %userprofile%\goto_ooRexx32
      %userprofile%\setup_environment4ooRexx64
      %userprofile%\setup_environment4ooRexx32
Demonstration

• In this case a Windows 10 machine
  – Demonstrating from USB stick
    • Creating the scripts
    • Running off the USB stick
  – Demonstrating copying the content of the "ooRexx USB stick" to the file system
    • Creating the scripts
    • Running scripts off the file system
Roundup and Outlook

- Includes currently five ooRexx interpreters
  - Standardized directory layout
- Scripts that are easy to use
  - Can use ooRexx off an USB stick
  - Can use ooRexx off a copy in the filesystem
- Possible future improvements
  - Extendable!
    - E.g., "arm"-Linux versions, once available
    - Enhance, e.g., with BSF4ooRexx