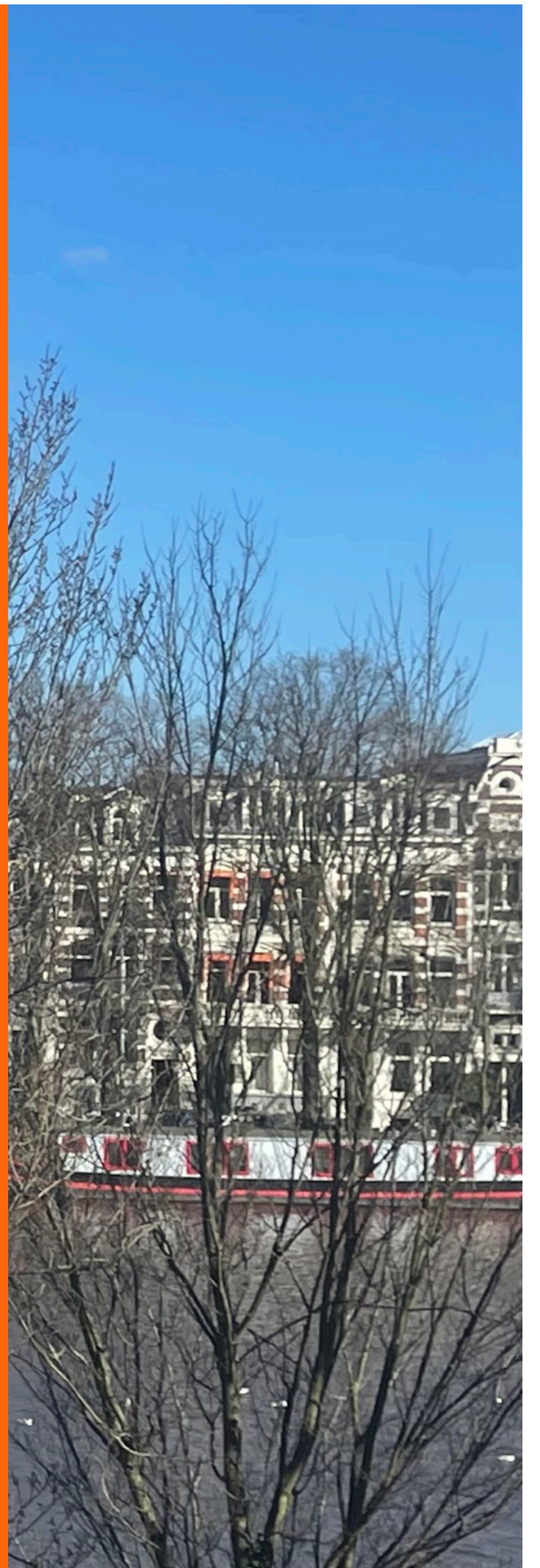
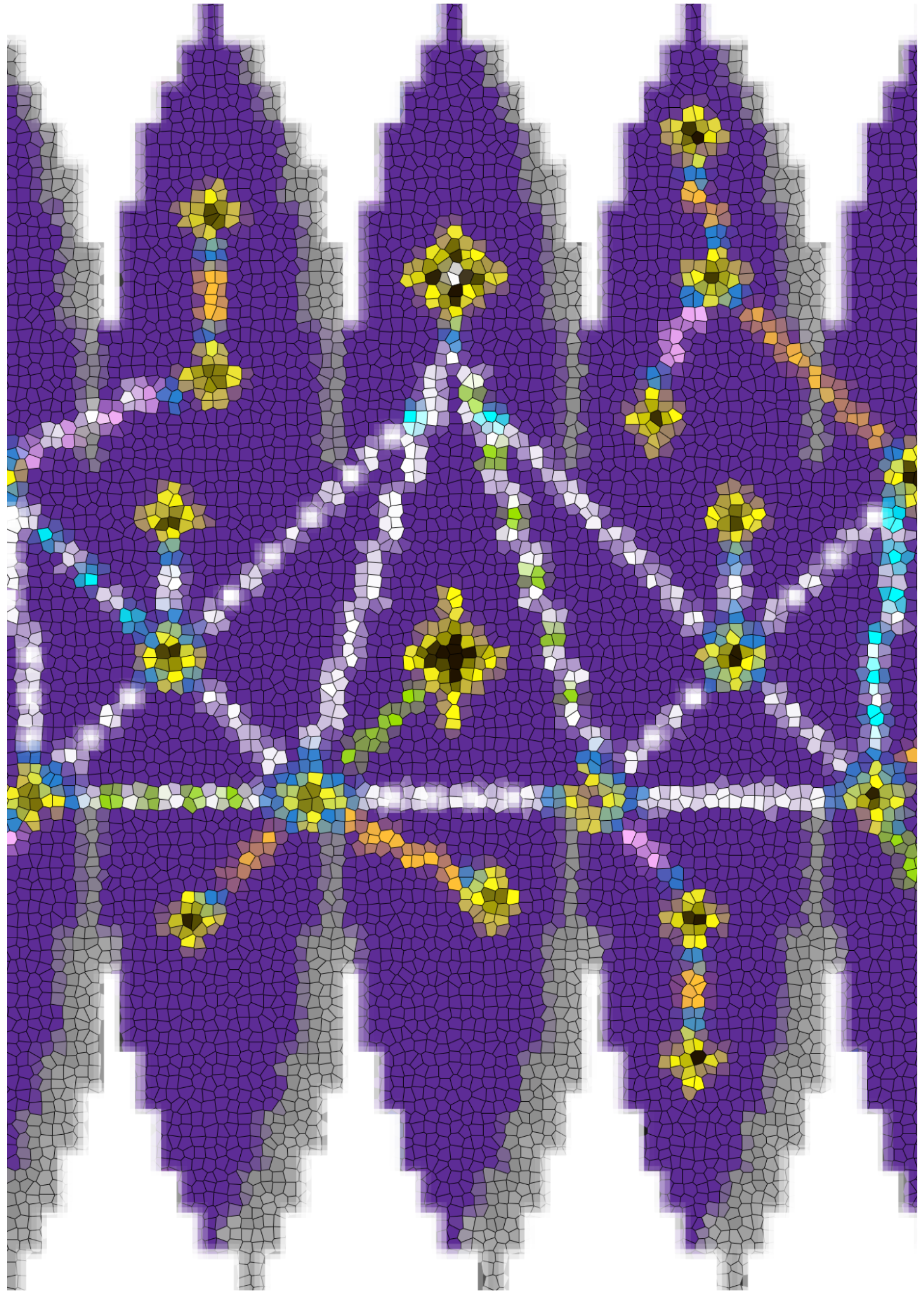




NetRexx 4.04 GA

René Vincent Jansen
33rd Rexx Language Symposium
12 September 2022





Our 4.04 release is more **future proof**, and at the same time more **compatible with the past** than it ever was

Agenda

- 1 Project Developments
- 2 Translator Developments
- 3 Workspace Developments
- 4 Pipelines Developments
- 5 Documentation and Examples

Nº 1

Project Developments

Project Location

- Issue List has moved to Github
 - After being in disuse since Oracle shut down our previous forge
 - SourceForge has a finite future for the RexxLA NetRexx project
 - Among other reasons, instability and taking our list addresses hostage
 - <https://github.com/RexxLA/NetRexx/issues>
- This repository is also a shadow location for the source code
 - <https://github.com/RexxLA/NetRexx>

Search or jump to... Pull requests Issues Marketplac

RexxLA / NetRexx Public

<> Code Issues 32 Pull requests Actions Projects

Filters is:issue is:open

32 Open ✓ 3 Closed

- Rexx Date preferred over Java Date and not intende
#35 opened 29 days ago by rvjansen
- Concatenating null values
#34 opened 29 days ago by rvjansen
- Exception when interpreting code with constant pro
#33 opened on 28 Jul by remesm
- XLATE bug with WordSeparator input range
#32 opened on 28 Jun by Jeff-Ft-Myers
- address with issue in pipelines
#31 opened on 22 Jun by rvjansen
- xlate issue in pipelines
#30 opened on 22 Jun by rvjansen
- Date('n',2022166,'j') gives incorrect output
#29 opened on 15 Jun by rvjansen
- parse source should give indication of current addr



959 lines (752 sloc) | 61.6 KB

<> Raw Blame

NetRexx 4.04 GA [12 Sep 2022]

Project

- the NetRexx issue tracker has moved to [<https://github.com/RexxLA/NetRexx/issues>]
- The RexxLA/NetRexx shadow repository will be maintained to be identical in contents to the SourceForge one

Language

- non-oo notation for scripting mode (René Vincent Jansen) enables (much more) Classic Rexx source compatible scripting
- a new nr command for scripting purposes
- releasenotes are in Markdown syntax (and have moved to the root of the development tree)
- use of color in language processor messages enabled (but optional, opt-out for Linux and Mac, opt-in for windows)
- long and short form of options
- a bugfix on an address statement ("variable \$2 already defined") with interruptedexception
- add the netrex/lang/Sysutils class by Marc Remes

The release notes are in Markdown now



No 2

Translator Developments

Classic Rexx compatible BIF

- BIF: Built in Functions (substr(), length(), left(), right() *etcetera*).
- Implemented in the RexxRexx class (to be analogous with RexxDate, RexxStream and other compatibility efforts)
 - Automatically included in 'Scripting Mode'
 - In 'class mode': use 'uses'

```
File Edit Options Buffers Tools Index NetRexx Help
package netrexx.lang
options binary decimal strictargs strictcase noformat
/**
 * With this one weird trick ... Rexx is back in 1984
 * Class RexxRexx provides static versions of the Rexx
 * to be called without oo notation.
 * Created on: Fri, 01, Apr 2022 13:04:19 +0200
 */
class RexxRexx

method RexxRexx()

method abbrev(a=Rexx, b=Rexx, length='') static
if length='' then len=b.length()
else len=length
return a.abbrev(b,len)

method abs(a) static returns Rexx signals NumberForm
return a.abs()

method b2d(a=Rexx,dil=Rexx -1) static returns Rexx
return b2d(a,dil)

method b2x(a=Rexx) static returns Rexx
return a.b2x()

method centre(a=Rexx, wid=Rexx, pad=Rexx ' ') static
return a.centre(wid,pad)

method center(a=Rexx,wid=Rexx, pad=Rexx ' ') static
return a.center(wid,pad)

method changestr(a=Rexx,old=Rexx,new=Rexx) static re
return old.changestr(a,new)
```


Why? “It really was fine as it was ...”

- NetRexx was conceived as a fully object-oriented language, and method calls have the form ***[instance].[method](parameters)***
 - The instance points to where the non-oo (let's call it procedural) version has the object that needs to be processed by the function
- Over the years, users indicated that they would like to use older scripts that they had written over the years
 - '*Scripting mode*' (available from the early beginnings) is a nod to that requirement
- Users of simple pipelines especially balked at having to switch around the objects and arguments in their own pipeline stages - these users are only in it for the (mostly compatible) pipelines, not specifically the other NetRexx benefits
- This increases your chance of running an old (but really simple) script unchanged,
 - And greatly reduces the work to get a one-off running quickly
- (And the ultimate retort - you don't have to use it if you don't want to)

```
file: half.rexx
/* rexx */
parse arg number
if datatype(number,'whole')
then say number%2 + number //2
else say ' argument is not a whole number'
```


Static functions

- These implementations of the BIF's are **static** calls
 - Ironically, the greatest benefit is for those that are a bit unsure about the difference between instance methods and static methods
 - You cannot do the usual oo-magic here, but that is the whole point

```
File Edit Options Buffers Tools Index NetRexx Help
package netrexx.lang
options binary decimal strictargs strictcase noformat
/**
 * With this one weird trick ... Rexx is back in 1984
 * Class RexxRexx provides static versions of the Rexx
 * to be called without oo notation.
 * Created on: Fri, 01, Apr 2022 13:04:19 +0200
 */
class RexxRexx

method RexxRexx()

method abbrev(a=Rexx, b=Rexx, length='') static
  if length='' then len=b.length()
  else len=length
  return a.abbrev(b,len)

method abs(a) static returns Rexx signals NumberForm
  return a.abs()

method b2d(a=Rexx,dil=Rexx -1) static returns Rexx
  return b2d(a,dil)

method b2x(a=Rexx) static returns Rexx
  return a.b2x()

method centre(a=Rexx, wid=Rexx, pad=Rexx ' ') static
  return a.centre(wid,pad)

method center(a=Rexx,wid=Rexx, pad=Rexx ' ') static
  return a.center(wid,pad)

method changestr(a=Rexx,old=Rexx,new=Rexx) static re
  return old.changestr(a,new)
```


Use of Colour/Color

- The translator now uses ANSI colour sequences to indicate the difference between warnings and errors
- This is switched on as a default, except for Windows, where we need to wait until more people use a sensible shell
- There are options **-color** and **-nocolor**
 - When **-colour** and **-nocolour** are really needed, we'll see the votes on the issue and will do our homework



Some bug fixes

- a bugfix on an address statement ("variable \$2 already defined") with interruptedexception
- date('j') input format gave incorrect output and is fixed. This version 'century windows' around 75 for compatibility with z/VM CMS Rexx; in a coming release 2-digit years will be deprecated on date'j' input format.

Sysutils

- The SysUtils class by Marc Remes is added to the distribution
 - SysUtils is a traditional package that delivers
 - sleep()
 - I/O Functions
 - File search - search a path
 - File properties (datetime)
 - Create/remove directories
 - SysFileTree
 - Mutex Semaphores
 - Event Semaphores



NetRexx Trace Interactive

This deserves its own presentation, which Marc Remes will deliver

TRACE INT

Marc Remes

remesm@gmail.com



No 3

NetRexx Workspace Developments

What is new and/or fixed in the Workspace

- You can now exit it with the **exit** command
 - I know this isn't a great feat but you don't have to remember what was that again ...?
- The **timer** on the sessions has been fixed
- It now starts up with the timer option (right side of screen) as a default
 - This was the intention earlier but that first starting time was way off - so instead you were shown the session number (which was correct)
- the default prompt changed to 'Ready;' (but both options can still be changed in an \$HOME/nrws.properties file)
- the **pipelines processor** is now loaded by default

No 4 Pipelines Developments

Pipelines

- a pipelines example that uses the ADDRESS WITH functionality in a NetRexx script
- Stage **locate** can be abbreviate to locat, loc, lo and l

```
OUT = ''
ADDRESS PIPE WITH OUTPUT STEM OUT
"LITERAL AAP NOOT MIES | SPLIT | REVERSE | CONS"
DO I=1 TO OUT[0]
  SAY OUT[I]
END
```


No 5

Documentation and Examples

Documentation

- Programmer's Guide made up to date with currently included scripting environments
- New Date(), Time(), and Stream() (linein, line out, charin, charout) functions documented
- ADDRESS WITH documentation added
-

A wonderful Swing GUI Sample

Thanks to Ruurd Idenburg

```
emac 201
FILE EDIT OPTIONS BUFFERS TOOLS INDEX NETREXX HELP
CLASS BEZIERCURVES.DrawCANVAS DEPENDENT PRIVATE EXTENDS JPANEL -
  ADAPTER IMPLEMENTS MOUSELISTENER, MOUSEMOTIONLISTENER

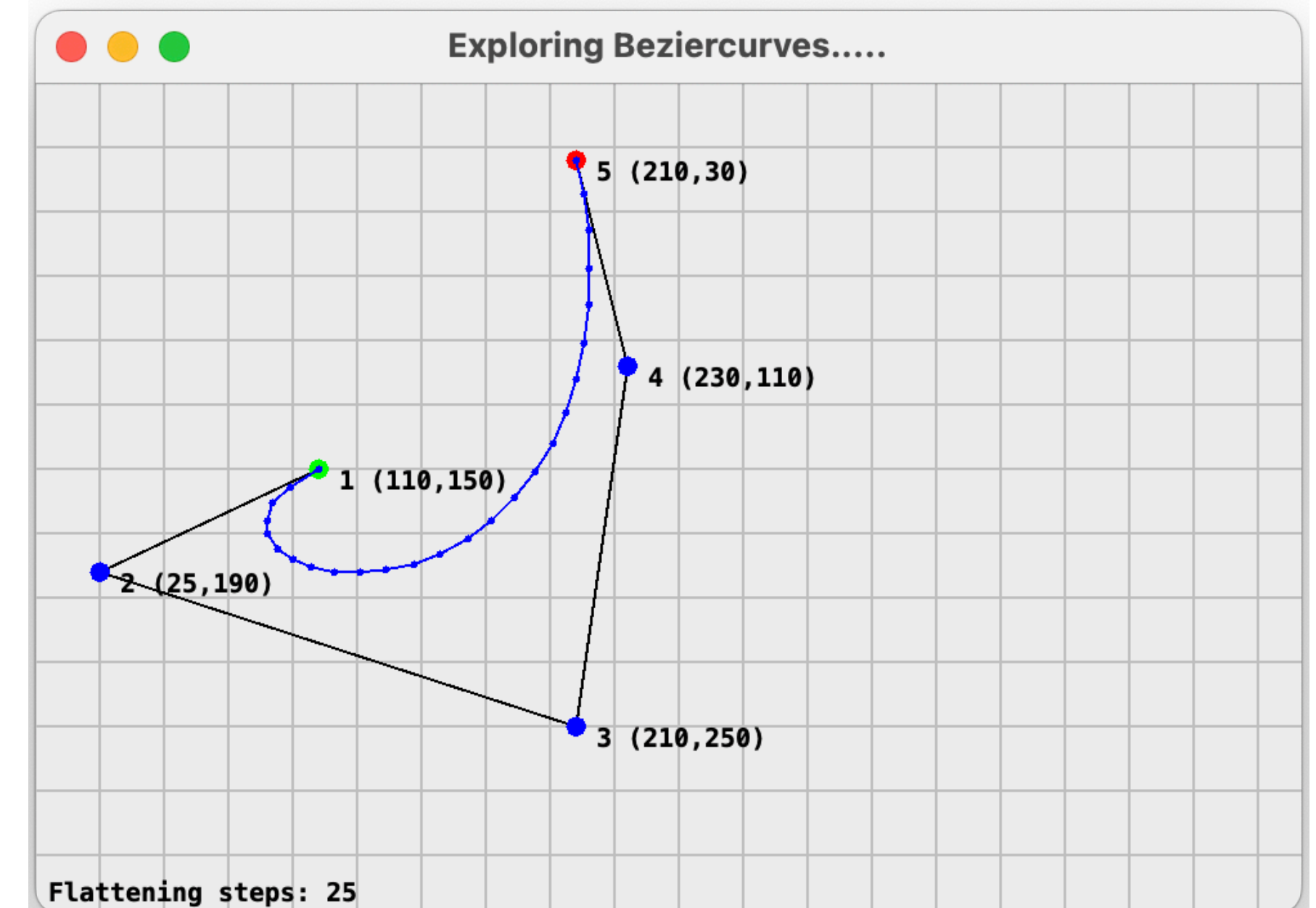
  PROPERTIES PUBLIC
  -- FIRST POINT IS START, LAST POINT IS END, IN BETWEEN ARE CONTROL
  POINTS=ARRAYLIST()
  -- CURVEPOINT IS UPDATED EVERY FLATTENING STEP
  CURVEPOINT
  -- SIZE OF A CURVEPOINT
  CURVEPOINTRADIUS = 5
  -- DISTANCE BETWEEN X AND ALSO Y AXES
  SKELETONAXESINCREMENT = 25
  -- THE NUMBER OF FLATTENING STEPS TO CALCULATE THE BEZIER CURVE
  FLATTENINGSTEPS = 10

  METHOD DrawCANVAS(STEPS)
  -- CREATE THE INITIAL START-, END- AND CONTROL-POINTS
  POINTS.ADD(110 150)
  POINTS.ADD(25 190)
  POINTS.ADD(210 250)
  POINTS.ADD(230 110)
  POINTS.ADD(210 30)
  -- SET THE NUMBER OF FLATTENING STEPS
  FLATTENINGSTEPS = STEPS
  -- SET THE MOUSE LISTENERS WE NEED AND IMPLEMENT
  ADDMOUSELISTENER(THIS)
  ADDMOUSEMOTIONLISTENER(THIS)

  METHOD MOUSEDRAGGED(EVENT=MOUSEEVENT)
  MOVECURVEPOINT(EVENT)

  METHOD MOUSERELEASED(EVENT=MOUSEEVENT)
  MOVECURVEPOINT(EVENT)

  METHOD MOVECURVEPOINT(EVENT = MOUSEEVENT)
  EX = EVENT.GETX
-UUU:----F1 BEZIERCURVES.NRX 26% L71 GIT-MASTER (NETREXX) -----
```



No 6

Recap of the 4.03 changes

(4.03 was released after the 2021 Symposium)

NetRexx 4.03 changes

- - ADDRESS WITH implemented (Marc Remes)
- - a fix for NETREXX-24 (was 148): ITERATE without label is not inside LOOP construct message (Marc Remes)
- - a small fix for the MOD option (Jason Martin)
- - correct slightly erroneous DataIO2.nrx example from NrxRedBk (René Vincent Jansen) (Thank you Terry Fuller)
- - a fix for JavaFX usage with the JPMS (and any other imported module) (Marc Remes) (Thank you Terry Fuller)
- - a new sample how to address https by Ruurd J. Idenburg
- - A Classic Rexx inspired Stream I/O implementation (René V. Jansen)
- (Thank you Leslie Turiff for testing and issue reports)
- - A new example of how to use a Swing Gui by Ruurd J. Idenburg
- - Start of a new organisation of the examples directory by function
- - a 'scripting mode' application adds 'uses RexxDate, RexxStream' to the generated class definition
- - RxJrt now compiles under Java 8 and RxJrtApi.nrx under Java >=9 (Marc Remes)
- - The java2nrx tool has been upgraded to v1.05
- - The distribution package now also contains java2nrx (previously only in source package)
- - in scripting mode, 'parse pull' can be used as an equivalent of 'parse ask'

No 7

Next: 4.05

NetRexx 4.05

- 📦 **Your suggestions, please**

- 📦 Documentation check and update

- 📦 Might contain

- 📦 Closer integration of Rexx and String types

- 📦 Calling natively implemented Rexx libraries

- 📦 Thank you for your attention!