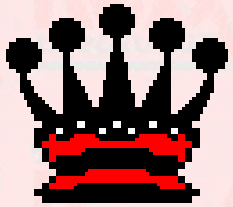
Regina Rexx

and other cool features

Florian Große-Coosmann & Mark Hessling

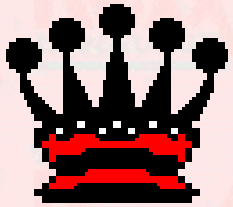
2004 Rexx Symposium
Böblingen-Sindelfingen, Germany
May 3, 2004

(Thanks to **imc** for background image)



Overview

- ANSI features
- OPTIONS
- Other interpreter features
- API additions
- Other platforms



ANSI features

- ADDRESS environments
- LOSTDIGITS
- .RS, .RC, .RESULT variables
- LINES(,'C'), CHARS(,'C')

ADDRESS environments

`ADDRESS` *env command* [*IO redir*]

env is one of

- `SYSTEM` or `OS2ENVIRONMENT` or `ENVIRONENT`
- `PATH` or `CMD` or `COMMAND`
- `REGINA` or `REXX`

ADDRESS REXX

`ADDRESS REXX` *scriptname* [*args*] [*IOredir*]

- SYSTEM INDEPENDENT
- INTERPRETER INDEPENDENT

nearly equivalent to

`ADDRESS PATH "rexx"` *scriptname...*

but the same execution program is used if the path can be determined.

ADDRESS ...WITH

IOredir = `WITH` [`INPUT` *in*] [`OUTPUT` *out*] [`ERROR` *out*]

```
in    =  NORMAL | IO
out   =  NORMAL | [APPEND | REPLACE] IO
IO    =  STEM symbol
      |  STREAM symbol
      |  STREAM string                 /* ANSI extension */
      |  LIFO string_or_symbol         /* ANSI extension */
      |  FIFO string_or_symbol         /* ANSI extension */
```

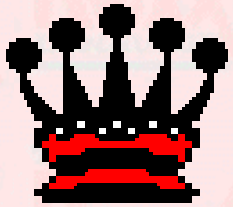
""-strings are equivalent to standard IO or current queue.

External queues are supported.

ADDRESS speed comparisons

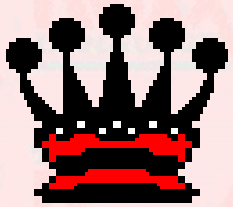
118 files, 137836 lines, 4195774 chars (machine: 2*350MHz PentiumII)

```
------------- executed all files at once --------------------
cat >/dev/null                                  =   0.067
cat | cat >/dev/null                            =   0.102
cat | ADDRESS STREAM | cat >/dev/null           =   0.289
cat | ADDRESS STREAM w/IO | cat >/dev/null      =   6.847
cat | ADDRESS STEM | cat >/dev/null             =  57.684
cat | ADDRESS FIFO | cat >/dev/null             =   1.651
---------- executed individually, summarised -------------
cat >/dev/null (using ADDRESS SYSTEM)           =   4.766
cat >/dev/null (using ADDRESS PATH WITH)        =   1.873
cat >/dev/null (using ADDRESS SYSTEM WITH)      =   4.515
catb>/dev/null (using ADDRESS SYSTEM)           =   4.530
catb>/dev/null (using ADDRESS PATH WITH)        =   1.866
catb>/dev/null (using ADDRESS SYSTEM WITH)      =   4.482
cat | cat >/dev/null                            =   5.171
cat | ADDRESS STREAM | cat >/dev/null           =   9.386
cat | ADDRESS STREAM w/IO | cat >/dev/null      =  16.128
cat | ADDRESS STEM | cat >/dev/null             =  18.993
cat | ADDRESS FIFO | cat >/dev/null             =  16.018
```



ADDRESS quotation

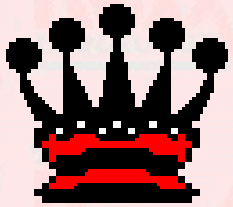
- ADDRESS PATH and ADDRESS REXX emulate ADDRESS SYSTEM's quotation rules
- double quotes and single quotes group words (default is whitespace)
- system depending escape character is used
  - "^" for OS/2 & Win32
  - "\" otherwise



ADDRESS signals

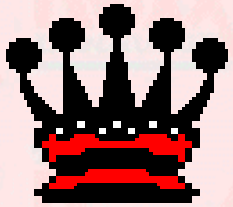
- conditions except syntax errors are raised only if enabled by SIGNAL
- FAILURE is raised if the desired program could not be loaded
- ERROR is raised if the called program terminates without success

- NOTREADY ***is not*** raised for streams
- NOVALUE ***is*** raised for stems
- SYNTAX is raised always for wrong values



External call signals

- conditions are raised only if enabled by SIGNAL
- FAILURE is raised if the desired script could not be loaded
- ERROR is raised if the called script terminates without success

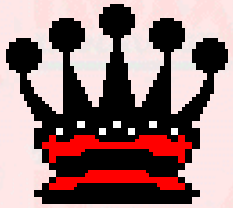


LOSTDIGITS condition

- LOSTDIGITS are not raised by default
- LOSTDIGITS are raised when a math op needs to truncate digits in the input where math op is
  - +
  - -
  - /
  - *
  - **
  - //
  - %
  - loop iterations

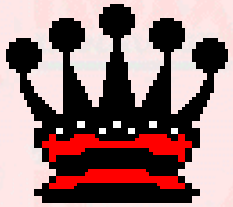
.variables

- Regina supports a separate pool for .vars
- access by name always uses "pool 0"
- Current variables are
  - .RC — copy of RC
  - .RESULT — copy of RESULT
  - .SIGL — copy of SIGL
  - .RS — -1=FAILURE, 1=ERROR, 0=success when using ADDRESS or ext. functions
  - .MN — x or x.x for syntax error numbers



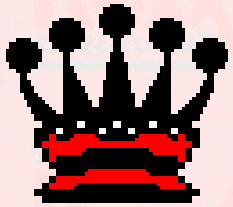
Variable pools

- hierarchically structured
- one pool structure per thread
- pool 0 contains dot-variables
- BIF POOLID() returns current pool id
- BIF VALUE() accepts a valid pool id as third parameter



LINES(,'C') and CHARS(,'C')

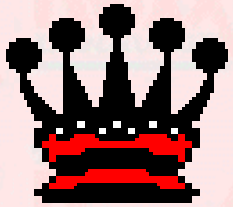
- LINES() and CHARS() take optionally 2 arguments
  - Stream name (as usual)
  - OPTION
    - 'N' default – return 0, 1 or actual number
    - 'C' count – returns actual number of lines/chars remaining in stream



OPTIONS

- STRICT_ANSI
- QUEUES_301
- TRACE_HTML
- STDOUT_FOR_STDERR
- CALLS_AS_FUNCS
- Environment variable REGINA_OPTIONS

M



Other interpreter features

- Error messages and Locale
- External queues
- PARSE LOWER and CASELESS
- TIME('T') and DATE('T')
- ARexx functions
- Regina invocation

Error Messages

- Support ANSI feature of ERRORTEXT BIF
  - ERRORTEXT(num, 'S')
    - Where num can be x.y
- Implemented with binary message files
- REGINA_LANG specifies language and optionally locale
- REGINA_LANG_DIR specifies location
- Regina Translation Project: http://www.bn.pl/~bk/serv/rrtp/index.rsp

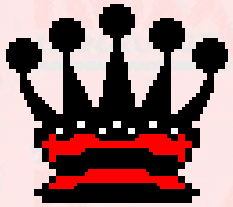
Locale support

- Locale support for whitespace and character translations like TRANSLATE(), UPPER() and LOWER()
- Consistency with locale irrespective of external influences
- Planning implementation for future enhancements (DATE, TIME, FORMAT BIFs)
- Implications of enhancements

M

External Queues

- Used to communicate between processes on any machine
- Queue name: name[@machine[:port]]
- Internal queues have no '@'
- Insecure – no access control
- Supported in RXQUEUE(), ADDRESS...WITH, RXQUEUE executable



Parse Lower and Caseless

- PARSE LOWER similar semantics to PARSE UPPER
- PARSE CASELESS allows strings in parse template to be matched irrespective of case
- CASELESS can be used with UPPER or LOWER

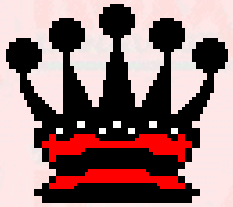
TIME(T) and DATE(T)

- 'T' option is for Unix time_t
- The number of seconds since 1 Jan 1970
- Advantage of date and time together
- NUMERIC DIGITS 10 minimum

ARexx Functions

- Introduced in 3.1
- Implemented by Staf Verhaegen
- Most BIFs implemented
- Requires OPTIONS AREXX_BIFS on platforms other than AmigaDOS or AROS

M



Regina invocation

- Run as Subroutine (-a)
- Tokenising (-c, -e)
- Safe Rexx (-r)
  - Can also be set in RexxStart() API with RXRESTRICTED in CallType arg
- Pause at end (-p)

REGINA_SUFFIXES

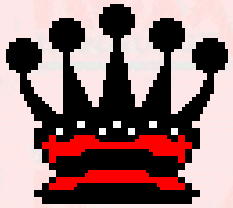
- Known suffixes
  - rexx
  - rex
  - cmd
  - rx
- Environment variable REGINA_SUFFIXES has form
  - suffix_w/o_dot[,suffix_w/o_dot]...
  - e.g. REGINA_SUFFIXES=ipret,regina

REGINA_SUFFIXES Algorithm

- if path given, look there
- if REGINA_MACROS given, look there
- if not "root" look in "."
- if PATH given, look there

- first found file wins always
- PATH-paths search without extensions
- known suffix forces other suffixes to ignore
- try REGINA_SUFFIXES
- try known suffixes

planned suffix "rxc" for precompiled scripts



Threading Model

- each thread has its own set of
  - variables
  - hooks
  - files
  - state
- ORexx (and Classic Rexx) and Regina have a different approach
- Environment variables are shared

API Additions

- ReginaCleanup()
  - Cleans up an interpreter instance
- RexxAllocateMemory()
  - Allocate memory passed to interpreter
- RexxFreeMemory()
  - Free memory allocated by interpreter
- RexxCallBack()
  - Call back into the running Rexx program

ReginaCleanup()

- Allows the API to reset the complete Rexx interpreter on a per-thread basis
- Variable pools are reset
- External libraries will be unlinked

RexxAllocateMemory()

- Move interpreter or platform-specific code into interpreter

RexxAllocateMemory() replaces code like:

```
#if defined(WIN32)
ret = ( char * )GlobalLock( GlobalAlloc (GMEM_FIXED, len));
#elif defined(OS2)
if ((BOOL)DosAllocMem((void **)&ret, len, fPERM|PAG_COMMIT ) )
   ret = (char *)NULL;
#else
ret = (char *)malloc(len);
#endif
```

with:

```
ret = (char *)RexxAllocateMemory(len);
```

RexxFreeMemory()

- Move interpreter or platform-specific code into interpreter

RexxFreeMemory() replaces code like:

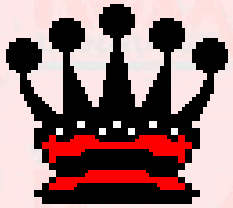
```
#if defined(WIN32)
GlobalFree( shv.shvvalue.strptr );
#elif defined(OS2)
DosFreeMem( shv.shvvalue.strptr );
#else
free( shv.shvvalue.strptr );
#endif
```

with:

```
RexxFreeMemory( shv.shvvalue.strptr );
```

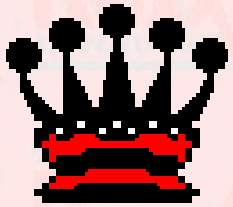
RexxCallBack()

- Allows the API to execute a Rexx procedure inside the running program
- Necessary for event-driven libraries like GUI interfaces
- Parameter passing with no interpretation
- Procedure executed within context of event handler



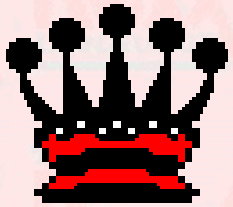
RxFuncErrMsg()

- Returns the actual error that caused RxFuncAdd() to fail
- More meaningful than a non-zero return code from RxFuncAdd()



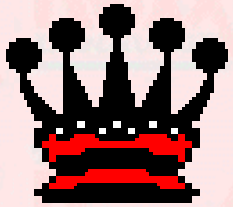
Other platforms

- Regina is the standard interpreter for:
  - AROS
    - http://www.aros.org
  - OSFree
    - http://www.osfree.org



Resources

- Regina home page:
  - http://regina-rexx.sourceforge.net
- API programming examples:
  - Available from SourceForge downloads page in *regina-documentation* subproject



Questions

?

M