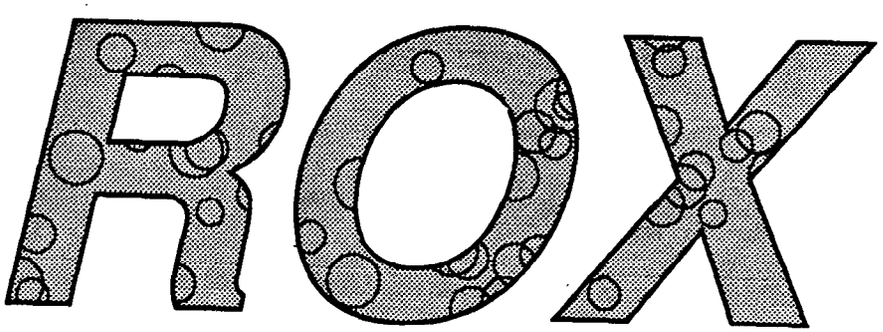
Adventures in Object-Oriented Programming in REXX

Patrick J. Mueller
IBM

Adventures in Object Oriented Programming with



(REXX Object eXtensions)

Patrick J. Mueller

pmuellr@vnet.ibm.com

May 1994, for the 1994 REXX Symposium

Trademarks

- IBM is a trademark of International Business Machines Corporation.

- OS/2 is a trademark of International Business Machines Corporation.

What is ROX?

- What ROX is:
  - A REXX function package for OS/2
  - Provides object oriented capabilities for REXX
  - An experiment
- What ROX isn't:
  - An interface to existing OO systems (C++, Smalltalk, SOM)
  - A new language
  - An IBM product

ROX object model

- Classes define:
  - Methods, implemented in REXX
  - Variables, accessible to methods
- Class inheritance
  - Classes obtain methods and variables of inherited classes
  - Multiple inheritance
- Modelled on Smalltalk, but:
  - Classes not 1st class objects
  - No garbage collection

Example ROX class

```
:*--------- animal class ----------
:class animal
:vars name sound

:method init
 name  = arg(1); sound = arg(2)

:method name
 return name

:method sound
 return sound

:*--------- dog class ----------
:class dog
:inherits animal

:method init
 name = arg(1)
 rc = animal.init(self,name,"Bark")
```

Example ROX usage

```
/* sample.cmd */

/* load the ROX file animal.rox */
rc = RoxLoad("animal.rox")

/* create a dog named Jackson */
dog = RoxCreate("dog","Jackson")

/* -> 'Jackson says Bark' */
say .name(dog) "says" .sound(dog)

/* destroy dog */
rc = RoxDestroy(dog)
```

Extra goodies

- C programming interface allowing methods to be implemented in C
- Auto-loaded DLLs to allow complete class definitions to be implemented in C
- Multithreaded support
- Execution profiling

Object creation/destruction

- Objects created with RoxCreate()
  - arg(1) is the class name
  - arg(2) ... are initialization parameters
  - The 'init' method of the class invoked automatically, if present
  - Initialization parameters passed to init method
- Objects destroyed with RoxDestroy()
  - The 'deinit' method of the class invoked automatically, if present

Object references

- RoxCreate() returns a string that is a reference to an object
- Object reference passed as first parameter to all methods, and RoxDestroy()
- Object references are plain old REXX strings - can be kept in a blank delimited string as in:

```
objs = ""
do i = 1 to 10
 objs = objs RoxCreate("dog")
end
```

- Special variables 'self' and 'super' available to methods which represent the receiver of the method

Sending messages

- Message sends are just REXX function invocations
- Object reference is always the first parameter
- Function name is method name, prefixed by "."
- Object and method name used to resolve the class that implements the method

  The two move methods invoked below are probably implemented in different classes:

```
xx = .add(aNumber,100)
xx = .add(aList,aListItem)
```

Instance variables

- Objects have as their instance variables all variables defined by their class, and its inherited classes.

- All instance variables apply only to a particular object - they are not shared between objects.

- All instance variables are 'exposed' when a method is invoked.

- Per-instance variables may be created with RoxAddVar(). This provides support for stemmed variables.

Packaging ROX classes

- RoxLoad utility allows classes to be packaged into their own files
- Multiple classes may be in one file
- Format is:

```
:include <a ROX file>

:class <class name>

:inherits <class name> ...

:vars <variable name> ...

:method <method name>
 <method code>

:method <method name>
 <method code>
```

Class-related functions

- RoxAddClass()
  create a class

- RoxClassAddInherit()
  add an inherited class to a class definition

- RoxClassAddMethod()
  add a method to a class definition

- RoxClassAddMethodDll()
  add a method (in a DLL) to a class definition

- RoxClassAddVar()
  add an instance variable to a class definition

Object-related functions

- RoxCreate()
  creates a new object
- RoxDestroy()
  destroys an object
- RoxSend()
  send a message to an object
- RoxSendThread()
  send a message to an object
  on another thread
- RoxClass()
  returns class of object
- RoxAddVar()
  add a per-instance variable
  to an object - used for stems

Utilities provided

- RoxLoad.cmd

  Calls the 'builtin' ROX functions to load a 'ROX' format file

- RoxInfo.cmd

  Prints class information for a given ROX file

- RoxProf.cmd

  Collects and analyzes output generated from RoxStats() function to generate timing information

Classes provided

- list.rox
- wordlist.rox
- set.rox
- collect.rox
  various collection classes;
  collect.rox is an abstract class

- sessions.rox
  illustrates multiple inheritance

- spinner.rox
  sample threaded class that displays
  an in-process spinner for activity

- cmdline.rox
  implements a function to read a line
  from input with history, editing, etc

- socket.rox
  usability enhancements for the
  rxSock function package

Problem areas

- Performance

  0.05-second overhead for message sends on 25/50 Mz 486 machine.

  That's pretty good, but still only 20 messages / second.

- File i/o

  Each invocation of a method opens a new file handle for a named file. Unpredictable because of buffering.

  Example: file 'a.file' opened twice

```
:method foo
 rc = lineout("a.file","x 1")

x = .foo(something)
x = .foo(something)
```

Implementation notes

- Uses REXX external function interface for message sends
- Internally, uses
  - RexxStart()
  - variable pool
  - init/term System exits
- Can be used by any REXX-macro-aware program
- Possible conflicts with programs that usurp REXX external function exit and depend on period prefixed functions

What's ROX good for?

- Experimenting with OO and REXX
- Whet your appetite for Object REXX
- A way to reuse large-ish chunks of REXX code, with shared variables

Availability

- Currently at version 1.8
- Available via:
  - anonymous ftp to ftp.cdrom.com in /pub/os2/program/rexx as rox.zip
  - Peter Norloff's OS/2 BBS

Availability

- Currently at version 1.8
- Available via:
  - anonymous ftp to ftp.cdrom.com in /pub/os2/program/rexx as rox.zip
  - Peter Norloff's OS/2 BBS