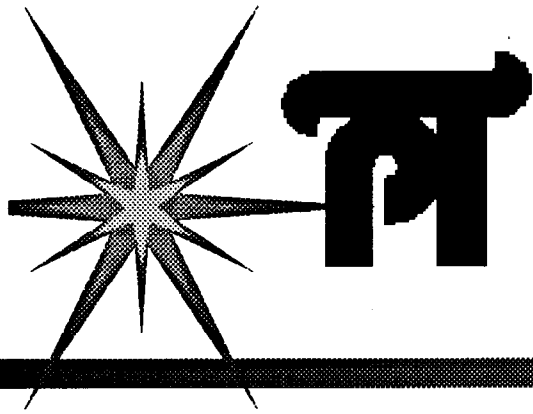# Problems and Issues Writing REXX Compilers

Markus Pelt-Layman
Pelt Industries

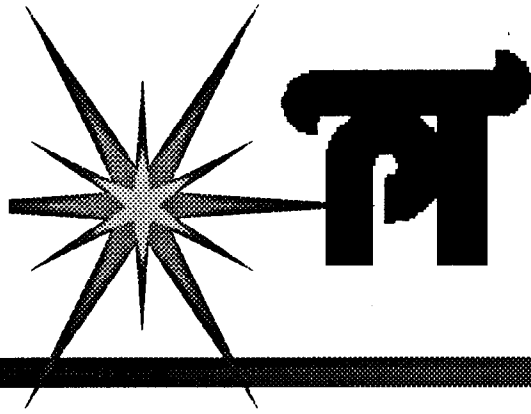**Pages 50-66**

50

Pelt Industries

REXXANNE™

# Problems and Issues writing REXX Compilers

Pelt Industries
History

- 1976 **TEACH** language interpreter (Honeywell)
- 1979 **DPS** command language (JES2 look-alike)
- 1985 **Intercept** aka **AF/Operator** command language
  (TSO CLIST language look-alike for operations automation)
- 1986 **OPS/MVS** REXX interpreter contract
  (ADDRESS ISPEXEC, automation rules, cmd/response API)
- 1987 Windows **Net*Edit** and **Elan Workstation** contracts
- 1989 OPS/MVS REXX **External Product Interface**
- 1992 OS/2 Smalltalk GUI front-end *(REXX EHLLAPI)*
- 1994 **REXXANNE** full-time
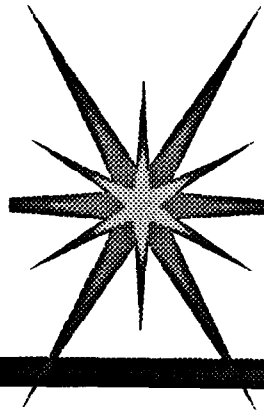
WHY REXXANNE ?

- ➤ I love REXX's simplicity and power
- ➤ Serious development requires a compiler
- ➤ I need
  - ◆ **Speed**
  - ◆ **Portability**
  - ◆ **High-level GUI framework**
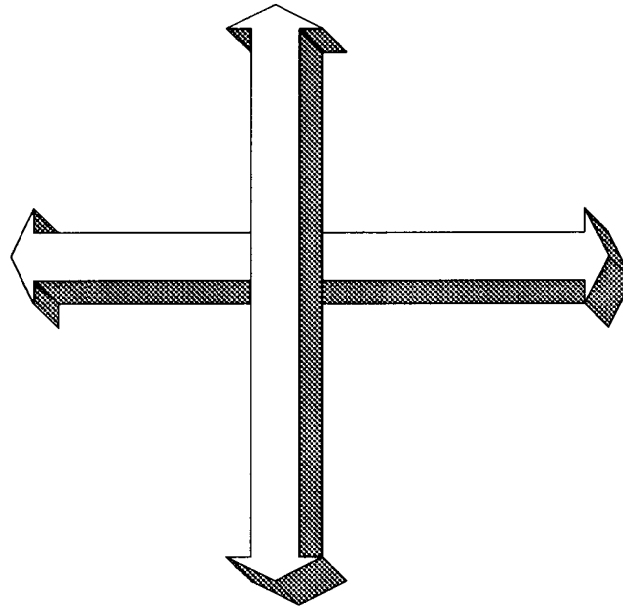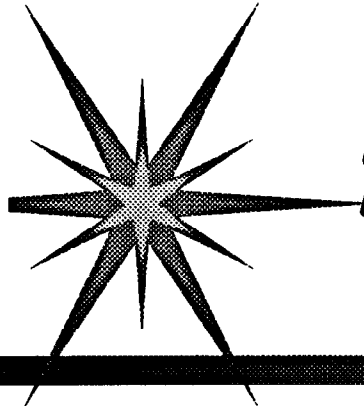
3
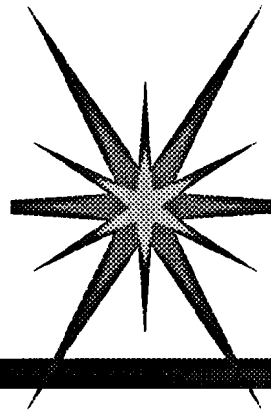
SOFTWARE FORCES

Speed

More hardware/
software platforms ⟷ Portability

More APIs

PORTABILITY

I   LANGUAGE DEFINITION

TRL/ANSI REXX/Object REXX

II   DEVELOPMENT ENVIRONMENT
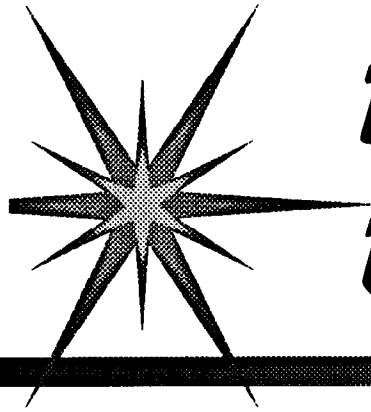
Portable interpreter/compiler, IDE

III   RUNTIME ENVIRONMENT

Portable run-time lib (built-in functions)
Portable ADDRESS environments (cmd/response)

IV   OBJECT CLASS HIERARCHY

Portable base class library, GUI class library

REXXANNE DESIGN GOALS

➤ Shorten development through portability

(no new learning on new platform)

- ◆ Cross-platform compiler
- ◆ Cross-platform IDE
- ◆ Cross-platform run-time library
- ◆ Cross-platform base class library
- ◆ Cross-platform GUI class library

➤ Shorten execution time
through compilation and optimization
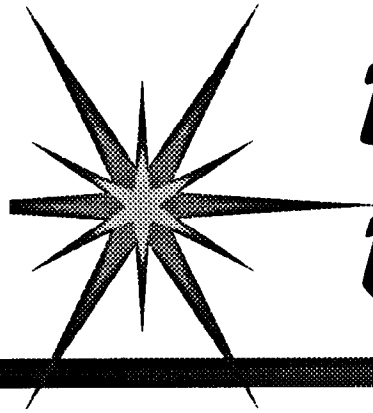
REXXANNE
DESIGN ISSUES

➤ Runtime licensing (INTERPRET)

➤ Run "interpreted" during development, compiled in production

➤ GUI portability versus platforms' native look and features (synthesis problem)

REXXANNE
CORE COMPILER

Version 1

C

```
Lexer
  └─ REXX tokens
Parser
  └─ P-code tokens
Emitter
  └─ Target .ASM code
Assembler
  └─ Target .OBJ code
Linker
  └─ .EXE .DLL or .COM
```

Version 2

C++

```
Lexer
  └─ REXX tokens
Parser
  └─ P-code tokens
Emitter
  └─ Triplets list
Optimizer
  └─ Triplets list (optimized)
Generator
  └─ Target .ASM code
Assembler
  └─ Target .OBJ code
Linker
  └─ .EXE .DLL or .COM
```

} New phases added

8
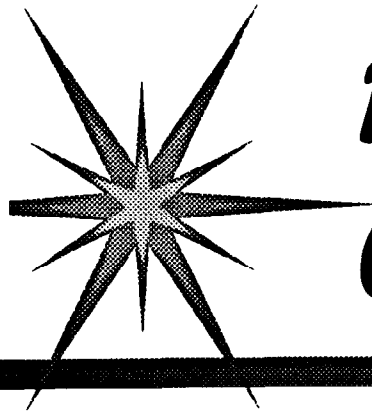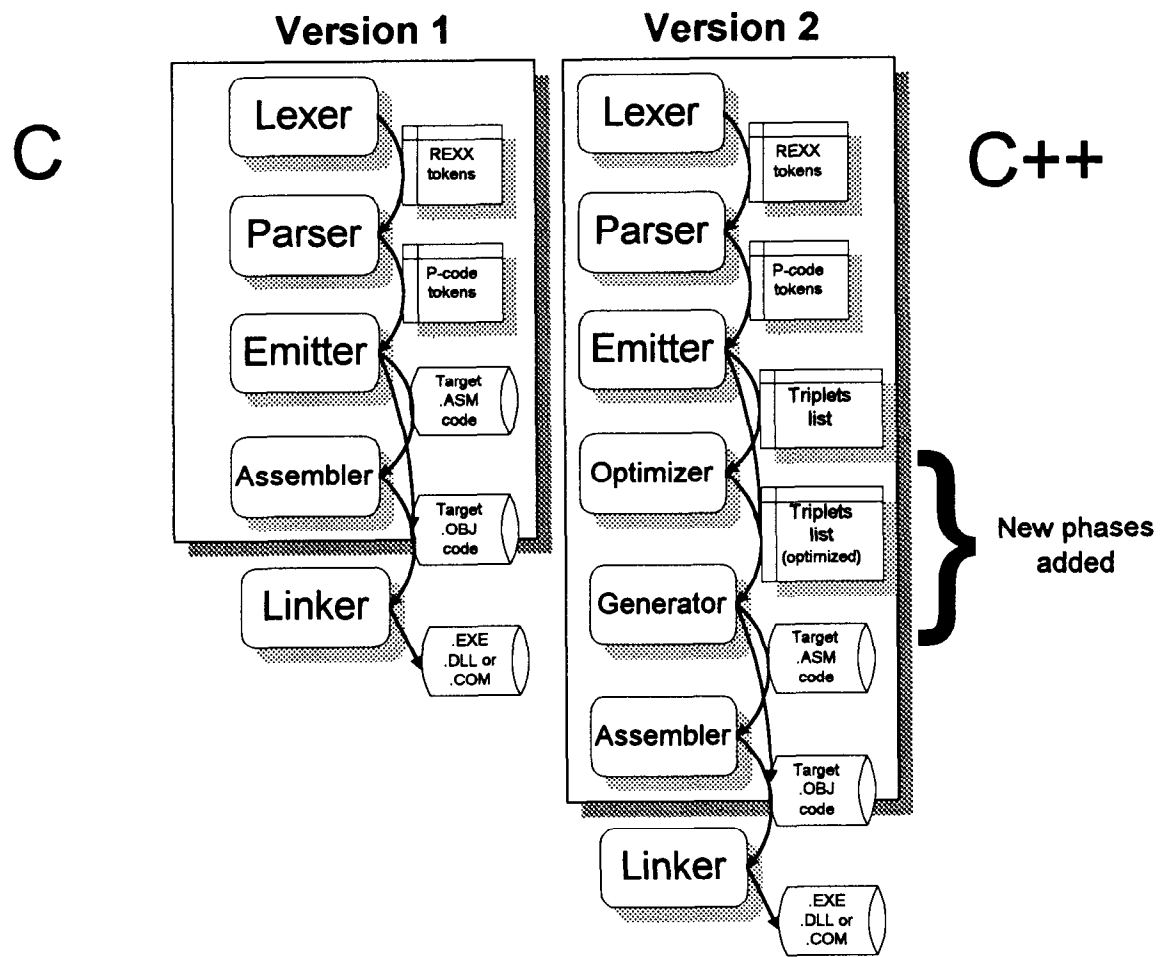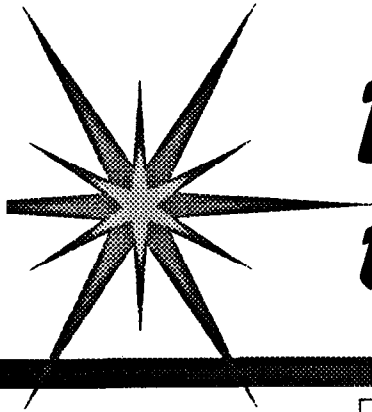
REXXANNE
DEVELOPMENT ENVIRONMENT

Pro Edition
Standard Edition
Princess Edition

REXXANNE Compiler
RXN

Editor

Debugger

IDE

LINKER
RXNLINK

Project Manager
RXNMAKE

Source Compare DIFF

Source Update UPDATE

Assembler **RXNASM**

Object Code Librarian **RXNLIB**

Class Hierarchy Browser

Legend

Description
Cmd line name

9
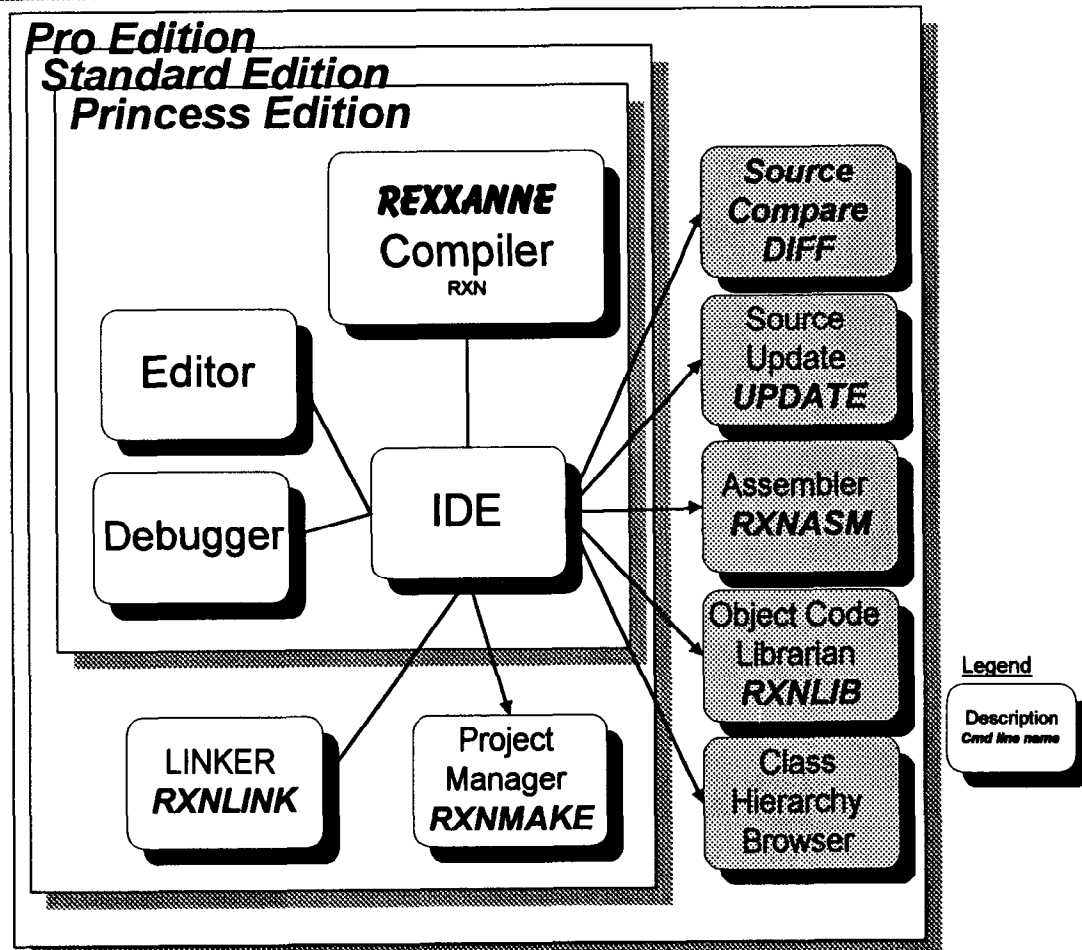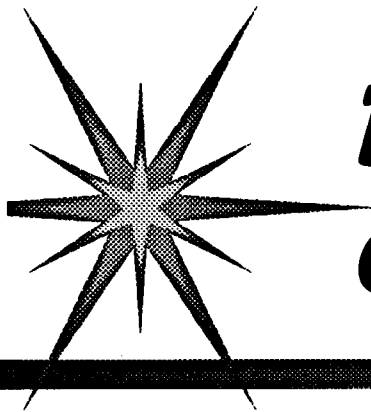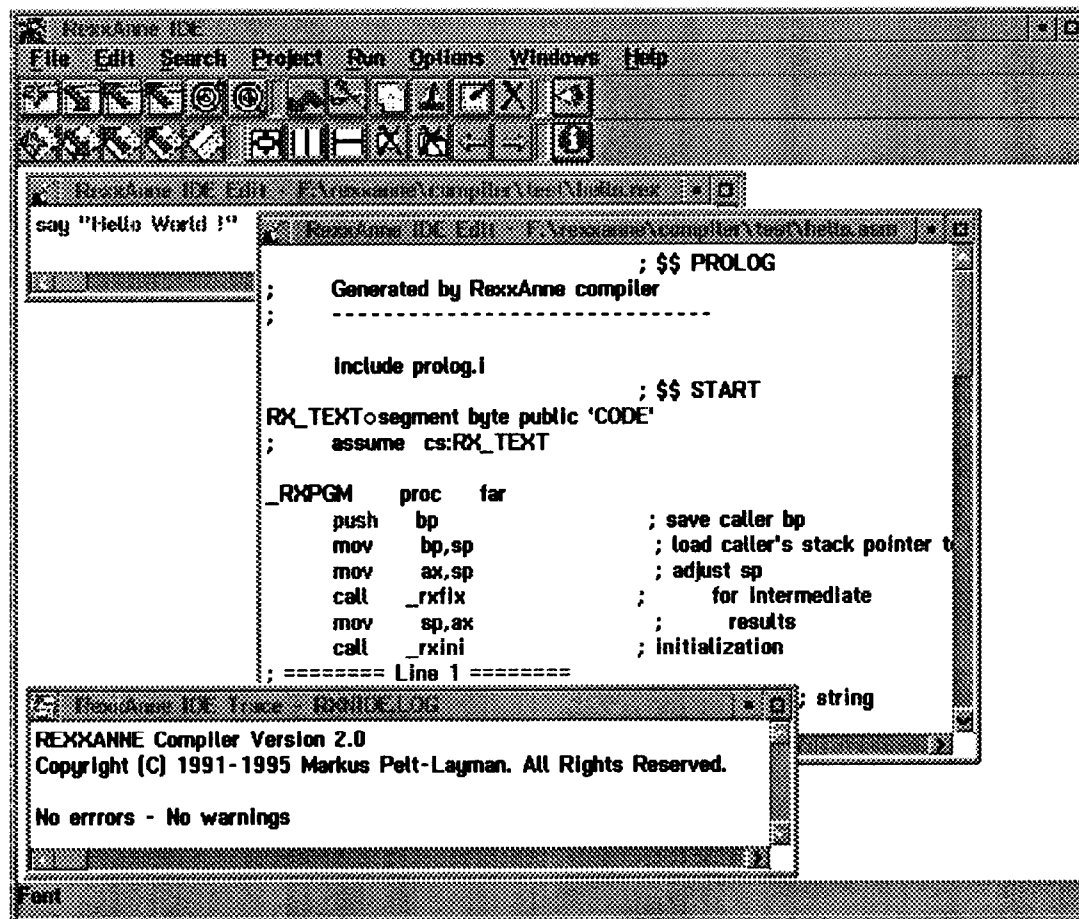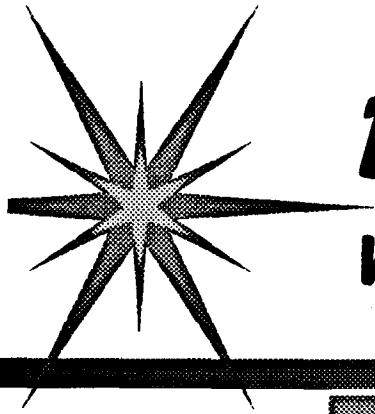
REXXANNE
OS/2 IDE

REXXANNE™
WINDOWS IDE

```
——                    RexxAnne/w - HELLO.ASM                    ▼ ▲
 Elle  Edit  Code  Project  Debug  View  Window              Help
 [□][▣][▣][▣][▣][▣][▣][▼]
——————————————————————— HELLO.REX ——————————————————  ▼ ▲
say "Hello World !"                                              ▲

  ——                          HELLO.ASM                      ▼ ▲
  ;   Generated by RexxAnne compiler
  ;   ----------------------------
  ;
      include prolog.i
  ;                              ; $$ START
  RX_TEXT segment byte public 'CODE'
  ;     assume  cs:RX_TEXT

  _RXPGM    proc   far
            push  bp              ; save caller bp
            mov   bp,sp           ; load caller's stack pointer to bp
            mov   ax,sp           ; adjust sp
            call  _rxfix          ;    for intermediate
            mov   sp,ax           ;       results
            call  _rxini          ; initialization
  ;========= Line 1 =========
            mov   bx,offset DGROUP:L1    ; $$ RVALUE; string
            PUSHARG ds
                                  ; $$ SAY1
            call  far ptr _RXSAY
            add   sp,tElemtSize
                                  ; $$ HALT
            mov   sp,bp           ; restore stack pointer
            pbp   bp              ; restore bp
            ret                   ; return
  _RXPGM    endp                  ; $$ END

  REXXANNE Compiler Version 2.0
  Copyright (C) 1991-1995 Markus Pelt-Layman. All Rights Reserved.

  No errors - No warnings
      Ready
```
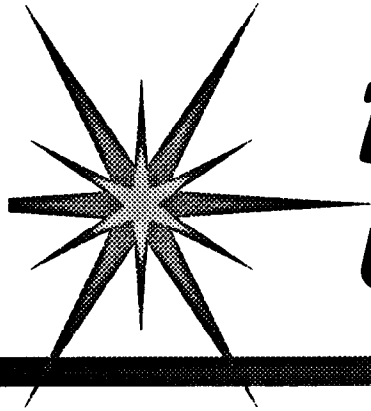
11
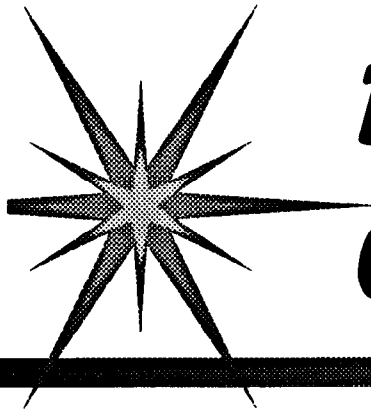
REXXANNE™
CURRENT STATUS

➤ We are **<u>WAY</u>** behind schedule

- ◆ under-staffed

- ◆ under-capitalized

- ◆ too ambitious a vision ?
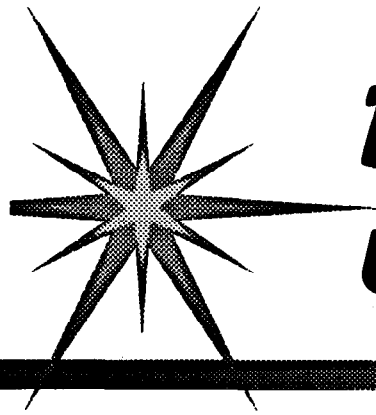
➤ Lack of user interest/support in REXX

REXXANNE
UNDERSTAFFING

➤ Task out work

    ◆ Lenny Koff - online tutorial & doc

    ◆ John Kastner - OS/2 IDE

    ◆ Billy Jack - runtime library

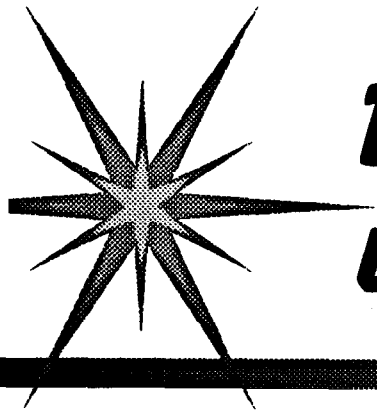   ◆ Find Windows/Windows NT/Windows 95 guru

➤ Cut down on vision

REXXANNE
UNDERCAPITALIZED

➤ Generate revenue

♦ INTER REXX subscriptions

♦ InfoREXX sales

♦ Enterprise REXX sales

♦ Miscellaneous REXX product sales

♦ Beta sales (free upgrade)

➤ Look for business partners

➤ Look for capital

REXXANNE

LACK OF USER INTEREST

➤ Generate interest in REXX language
- ◆ INTER REXX newsletter
- ◆ Get articles published in other magazines
- ◆ Prod IBM and others to better marketing

➤ Publish information on REXX
- ◆ InfoREXX multimedia help file
- ◆ Novice tutorial

➤ Generate interest in *REXXANNE*
- ◆ comp.lang.rexx newsgroup participation
- ◆ MSJ advertisements
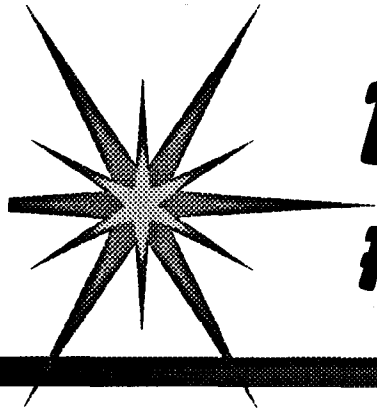
REXXANNE

FUTURE

- Continue REXX evangelism
- Release "early bird" version (free upgrade)
- Share development of new features with other vendors (IDE, APIs)
- Get users involved in development
- Continue long-term commitment to REXX