

News From the REXX Compiler

Klaus Hansjakob
IBM



Dr. Klaus Hansjakob

IBM Vienna Software Development Laboratory
Lassallestrasse 1
A-1020 Vienna, Austria
Europe

HANSJAKO@VABVM1.VNET.IBM.COM
ATIBMCXP at IBMAIL

(+431) 21145-4243

The information contained in this document has not been submitted to any formal IBM test and is distributed on an "As Is" basis without any warranty either expressed or implied. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

In this document, any references made to an IBM licensed program are not intended to state or imply that only IBM's licensed program may be used; any functionally equivalent program may be used instead.

Any performance data contained in this document was determined in a controlled environment, and therefore the results which may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

It is possible that this material may contain references to, or information about IBM products (machines and programs), programming or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such IBM products, programming or services in your country.

Agenda

- ④ ■ News from the REXX Compiler
 - Packaging an application
 - General considerations
 - DLINK
 - Function packages

Compiler

IBM Compiler and Library for SAA
REXX/370
Release 2

5695-013

5695-014

Available for CMS and MVS
Library is part of REXX/VSE

Alternate Library PTFs

	CMS		MVS	
	PTF	APAR	PTF	APAR
Compiler	UN51503 UN51504 JPN	PN48015	UN51833 UN51834 ENU UN51835 JPN	PN48006
Library	No PTF, additional product tape			

Introduction of Copyright, Alternate Library

Copyright



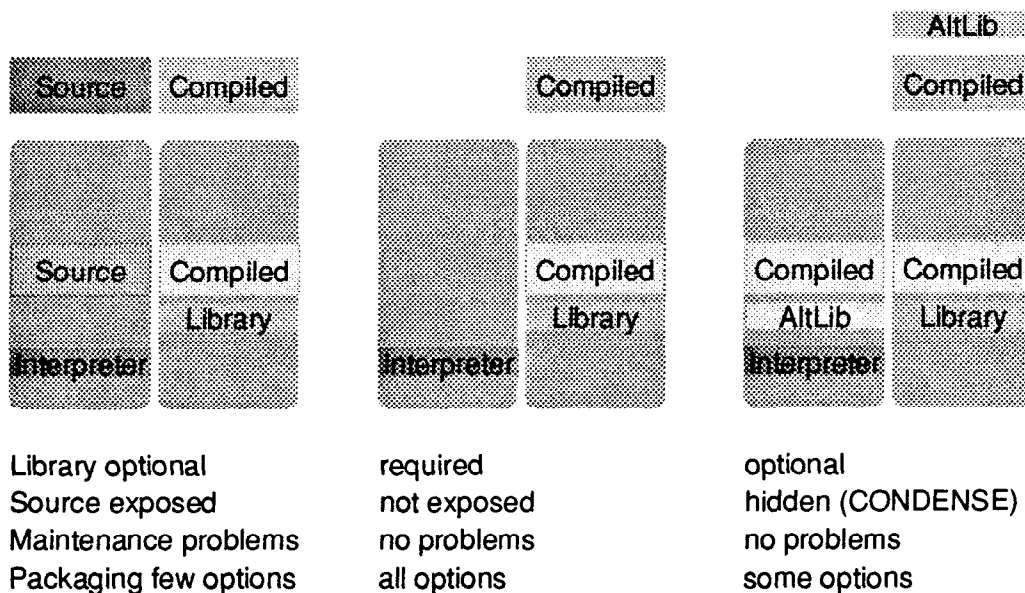
```
/* This program welcomes you ... */
/*%COPYRIGHT This program copyrighted for */
/*%COPYRIGHT MY Company, Vienna, Austria */
say 'hello world'
```

```
ESD HELLO a 00000001
TXT a @EXECPROCEAGRTPRC Compiled REXX 2.0 15 Mar 1994 100000002
TXT 3:46:29 CMS REXXC370 3.48 28 May 1993 PTF UN51503 c00000003
TXT u° ) -1^&& $j ^ a & & j ^a &µ - a0&i; 0 i0 00000004
TXT y o '10& a)&2 *j ^a &Q &o i ah- a10( a\0 00000005
TXT \ & aá & & q ) ú This program cop00000006
TXT yrighted for MY Company, Vienna, Austria 800000007
TXT & b ° e a + ( a É - 00000008
TXT h ~* à @ h m µ * é É A A ~ + 00000009
TXT ( / a a A - a f 0 Jê ç Ä é a ~00000010
TXT 8 : aá a a f : f é - J a 00000011
TXT ~: % H è AÆ J 1* f z ( N l ( á 0 10000012
TXT Ç ^ ( )o a^ 0 0 ; J%*j - f e l - -o 00000013
TXT µ l ö; A é' ^ H R µ ÉO )E K;°E &G °N &E jü É- 00000014
TXT aZ I / &U k ;°R &X C 7 . 8 2 \M y 1 °900000015
TXT Ék;°S I T °F °C ¥ éð *(0 A2 *1Z*f k ;é. )00000016
TXT c s °C &S L ÉH ~ Bè \ ¥a a* aç k ¥* K> 2 00000017
TXT é è éi é1 k: ¥@ K= 2 b d éf éh k« ¥1 á 0 2m 1á00000018
TXT ¼ j ) ° ^ bÆ 8 /Z B; ) J á ^ °è B¶ áÆ (00000019
TXT 0 u f á +; - / ° j ! / áI ó / M L j- -Ñ èà 00000020
TXT ~H i z /è a ~ B n 00000021
END 1569501301 010094074 00000022
```

Alternate Library

- Compile program
 - with ALTERNATE and SOURCELINE
 - use CONDENSE to hide source
 - DLINK does not work
- Distribute Alternate Library
 - without royalties, without paperwork
- Alternate Library
 - is installed on systems without Library
 - invokes interpreter when compiled program is run

Distributing REXX



What's missing

- TRACE support
- REXX I/O for CMS
- INCLUDE facility
- MARGINS

Agenda

- News from the REXX Compiler
- ① ▪ Packaging an application
 - General considerations
 - DLINK
 - Function packages

Performance

```

/* REXX Inner */
cvi=sysvar('SYSCPU')
Do i=1 To 1000
  Call inner
End
cve=sysvar('SYSCPU')
Say 'CPU time' cve-cvi
Exit

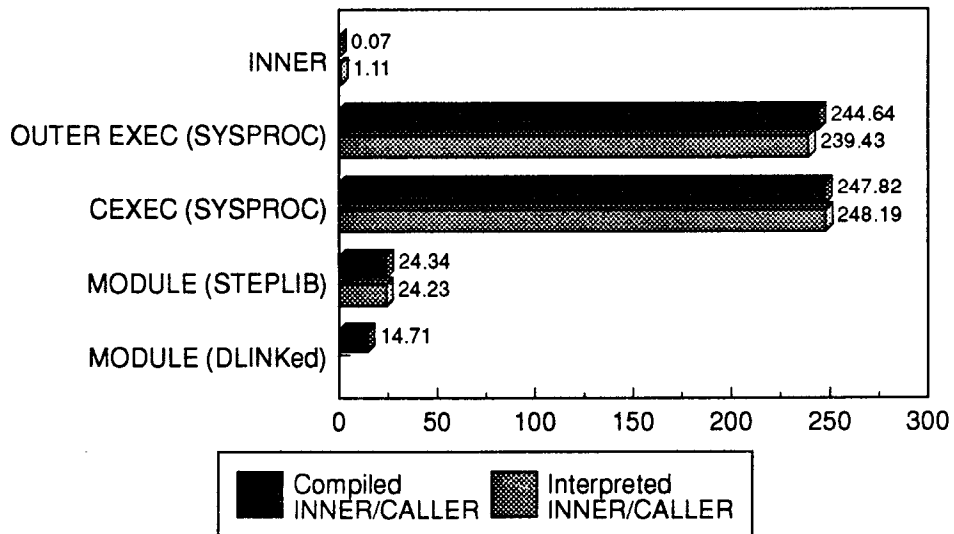
inner:Procedure
Return
  
```

```

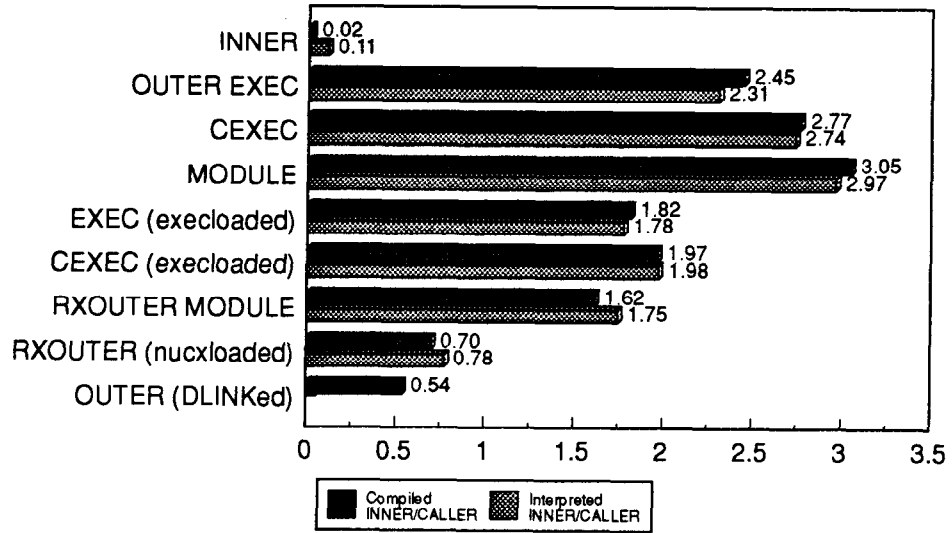
/* REXX Caller */
cvi=sysvar('SYSCPU')
Do i=1 to 1000
  Call outer
End
cve=sysvar('SYSCPU')
Say 'CPU time' cve-cvi
Exit

/* REXX Outer */
Return
  
```

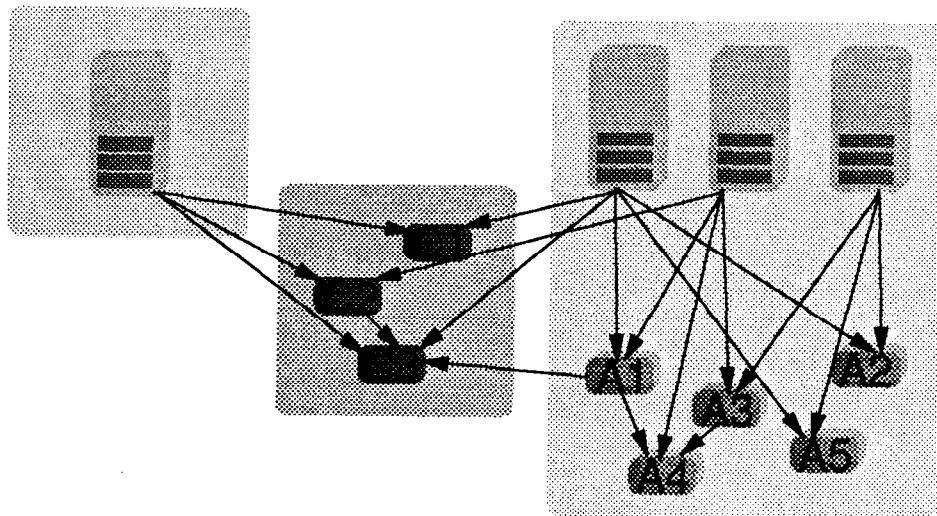
Performance (MVS)



Performance (CMS)

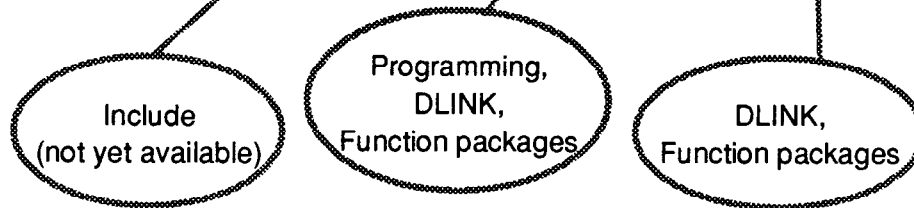


Applications

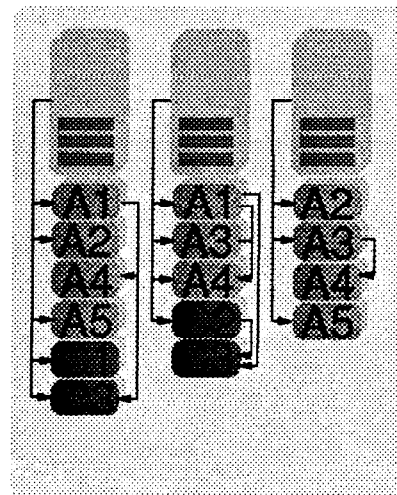
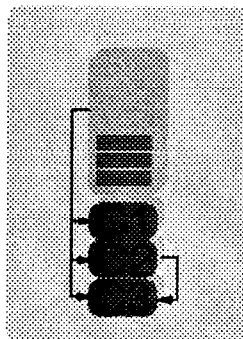


Internal - External

	Internal	External
Performance	good	bad
Maintenance	many	one
Distributed development	impossible	possible
Pieces	one	many
Search order	certain	uncertain
Variable sharing	easy	hard



DLINKed Applications



DLINK

- Search overhead zero
- Requires Compiler
- Does not work with Alternate Library

Function Packages

- Commonly used functions
- Early in the search order
- Functions must understand REXX function invocation

Search Order (CMS)

RXEXTFNC

RXSYSFN LOAD RXEXTFNC

RXLLOCFN LOAD RXEXTFNC

RXUSRFN LOAD RXEXTFNC

continue search

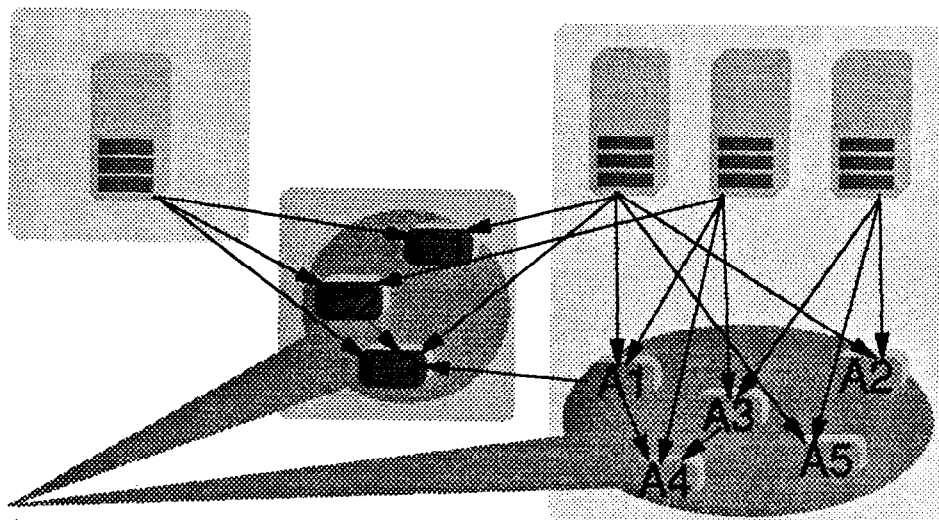
x=extfnc()
call extfnc

Make RXxxxFN a
nucleus extension

Make RXEXTFNC a
nucleus extension

invoke RXEXTFNC

Applications with Function Packages



Function packages

Function Packages

- First in search order
- Compiler allows to write functions in REXX
- Works with Alternate Library
- May require explicit loading/unloading on CMS
- DLINK may be used when Alternate Library is not used

CMS Function Package Example

- Two files
 - RXUSERFN is function package loader
 - USERFN is function package glue code
- "Glue" code for a function package
 - Use without royalties
 - Allows free naming of function package
 - Requires renaming of files
 - Explicit loading of package and all functions with "RXmyname LOAD"
 - Explained in RXUSERFN header

Necessary Modifications

RXUSERFN ASSEMBLE

&PACKAGE	SETC	'USERFN'	Name of the package to load
&RXPACK	SETC	'RX&PACKAGE'	Name of this program
&CR(1)	SETC	'My Copyright'	Copyright notice
&CR(2)	SETC	'second line'	Copyright notice continued

USERFN ASSEMBLE

&PACKAGE	SETC	'USERFN'	Name of the package
&CR(1)	SETC	'My Copyright'	Copyright notice
&CR(2)	SETC	'second line'	Copyright notice continued
&FUN(1)	SETC	'USER1'	Name of function
&FUN(2)	SETC	'USER2'	Name of function
&FUN(3)	SETC	'USER3'	Name of function

Obtain the Source Code

- Email - write a note to
 - hansjako@vabvm1.vnet.ibm.com
 - ATIBMCXP at IBMMAIL
- Disk - get one
 - as long as supply lasts
 - if you can't use email
 - if you have a way to upload code
- Supplement
 - type in

Agenda

- News from the REXX Compiler
 - Copyright
 - Alternate Library
- Packaging an application
 - General considerations
 - DLINK
 - Function packages
 - Function package example

News from the REXX Compiler - Supplement

Klaus Hansjakob

IBM Vienna Software Development Lab
Wien 2, Lassallestrasse 1

c/o IBM Austria
Obere Donaustrasse 95
A-1020 Austria
EUROPE

HANSJAKO@VABVM1.VNET.IBM.COM

(+431) 21145-4243

May/95

Function Packages



```

***** TITLE 'RXUSERFN * REXX Function Package Loading Stub' ***** 00010000
***** Describe your function package here ***** 00010100
* 00010200
* 00010300
* 00010400
* 00010500
* 00010600
* 00010700
* 00010800
* 00010900
* 00011000
* 00011100
* 00011200
*****Start of Specifications***** EJECT 00020000
* 00020000
* This code is provided on an as-is basis. * 00040000
* 00050000
***** 00060000
* 00070000
* Module name: RXUSERFN 00080000
* 00090000
* Descriptive name: REXX function package loader 00100000
* 00110000
* Function: 00120000
* 00130000
* NUCXLOAD module USERFN as RXUSERFN: when invoked with a LOAD 00140000
* request invoke USERFN with same PLIST as an entry. 00150000
* 00160000
* To generate: 00170000
* * HASM RXUSERFN 00180000
* * LOAD RXUSERFN ( ORIGIN TRANS 00190000
* * GENM RXUSERFN 00200000
* 00210000
* Note: RXUSERFN is interrogated as part of the REXX search 00210010
* order for external functions. The functions are loaded 00210020
* automatically when invoked. You can also create a function 00210030
* package with a different name (e.g. RXMYPKG): 00210040
* * Rename this file to RXMYPKG. 00210050
* * Change the macro variable PACKAGE in the header of this 00210060
* file from USERFN to MYPKG. 00210070
* * Rename the file containing the functions to MYPKG. 00210080
* * Change the macro variable PACKAGE in the header of the 00210090
* file containing the functions to MYPKG. 00210100
* * When the function package is not interrogated as part of 00210110
* the REXX search order you must load the functions in the 00210120
* package explicitly ('RXMYPKG LOAD RXname') where name 00210130
* is the name of the function) or globally ('RXMYPKG LOAD' 00210140
* before you can invoke them. 00210150
* * To drop all functions of a package issue a NUCXDROP command 00210160
* for the package ('NUCXDROP RXMYPKG'). 00210170
* 00220000

```

05/94 KH

News from the REXX Compiler - Supplement

1

Function Packages ...



```

* Entry/exit conditions: 00230000
* 00240000
* NOTE: The MODULE is generated as a transient module. 00250000
* 00260000
* Entry: 00270000
* Standard SVC conventions. 00280000
* RI points to a tokenized PLIST (SVC 202 linkage). 00290000
* 00300000
* Exit: 00310000
* * R15 = 0 USERFN successfully loaded and returned with 0 00320000
* * R15 = 4 - Return code from unsuccessful NUCXLOAD USERFN 00330000
* * - Return code passed back from USERFN after 00340000
* * invocation with original PLIST 00350000
* * - 4 to indicate bad PLIST 00360000
* 00370000
* 00380000
* Exit: 00390000
* Return to caller. 00400000
* 00410000
* Operation: 00420000
* When invoked without argument NUCXLOAD USERFN as RXUSERFN, 00430000
* and pass back the return code obtained from NUCXLOAD. 00440000
* When invoked with 'LOAD' as the first argument then NUCXLOAD 00450000
* USERFN as RXUSERFN, invoke RXUSERFN with the same PLIST as 00460000
* obtained on entry, pass back the return code given back by 00470000
* USERFN. 00480000
* Otherwise display message and return with return code 4. 00490000
* 00500000
* Macros: 00510000
* * CMSLIB or DMSGPI 00520000
* 00530000
* Macros and control blocks: 00540000
* * REGEQU 00550000
* 00560000
* Change Activity: 00570000
* * 91-11-21 KH Cleanup and comments 00580000
* * 93-06-28 KH Make function package name a macro variable, add 00590000
* * copyright as macro variable. 00600000
* *****End of Specifications***** 00610000
* 00620000
* BRXPack RMODE 24 Must be loaded below 16MB! 00630000
* BRXPack AMODE 24 Expects SVC 202 linkage 00640000
* 00650000
* BRXPack CSECT , 00660000
* * 00670000
* * Make sure this is a LOAD 00680000
* 00690000
* * USING * ,R12 Establish addressability 00700000
* * B STARTCOD Branch around header 00710000
* * DC CLB'BRXPack' Package ID 00720000
* * SETA N'&CR 00730000
* * SETA I 00740000

```

05/94 KH

News from the REXX Compiler - Supplement

2

Function Packages ...



```

.CRLOOP ANOP 00680000
* 00690000
* DC C'&CR(&I)' 00700000
* 00710000
* &I SETA &I+1 00720000
* AGO .CRLOOP 00730000
* .CRLOOPE ANOP 00740000
* 00750000
* STARTCOD DS 0F 00760000
* LR R10,R14 Save return address 00770000
* LR R2,R2 Assume install only 00780000
* CLI ARG(1),X'FF' Any arguments? 00790000
* BE GOLOAD Br if not - go install 00800000
* CLC ARG(0,R1),=CLB'LOAD' Is this explicit load? 00810000
* BNE BADPL Br if not - go complain 00820000
* LR R2,R1 Keep invocation PLIST 00830000
* 00840000
* * NUCXLOAD USERFN as RXUSERFN. 00850000
* 00860000
* GOLOAD EQU * 00870000
* LA R1,NUCXLOAD Address NUCXLOAD Plist 00880000
* SVC 202 00890000
* DC AL4(1) Return even if error 00900000
* LTR R15,R15 Did load work? 00910000
* BNZR R10 No, pass back rc 00920000
* * 00930000
* * If explicit load requested pass through invocation PLIST 00940000
* * 00950000
* * LTR R1,R2 explicit load? 00960000
* * BZR R10 No, return 00970000
* * SVC 202 Invoke nucleus extension 00980000
* * DC AL4(1) 00990000
* * BR R10 Return, pass through rc 01000000
* * 01010000
* * Error handling routines. 01020000
* * Note that in order to avoid the generation of relocatable 01030000
* * address constants, the TYPLIN PLIST is "hand-built" rather 01040000
* * than using WRITE. 01050000
* 01060000
* BADPL EQU * Something's wrong with PLIST 01070000
* LA R1,MSG1 Get message address 01080000
* LA R2,L'MSG1 Get message length 01090000
* STCM R1,0'0111',TYPBUFF Set it in PLIST 01100000
* STH 02,TYPLEN Set it in PLIST 01110000
* OI TYPLIN+13,X'40' Request error message edit 01120000
* LA R1,TYPLIN Point at PLIST 01130000
* SVC 202 Give it to CMS 01140000
* DC AL4(1) Ignore errors 01150000
* LA R15,4 Set non-zero return code 01160000
* BR R10 Return 01170000
* * 01180000
* * DROPT R12 01190000
* * 01200000
* TYPLIN DC CLB'TYPLIN',X'01',AL3(0),C'B',X'00',AL2(0)

```

05/94 KH

News from the REXX Compiler - Supplement

3

Function Packages ...



```

IYBPFF EQU TYPLIN+9,3          01130000
TYPLEN EQU TYPLIN+14,2       01140000
MSG1 DC C'DMSRUF070E Invalid parameter' 01150000
*                                     01160000
NUCXLOAD DS 00                01170000
DC CL8'NUCXLOAD'              01180000
DC CL8'&RXPACK'              01190000
DC CL8'&PACKAGE'             Name to load as 01200000
DC CL8'&MODULE'              Name of module to load 01210000
DC CL8'('                    01220000
DC CL8'&SYSTEM'              System attribute 01230000
DC CL8'&SERVICE'           Service attribute 01240000
DC CL8'&PUSH'               Push down existing extension 01250000
DC BX'FF'                   01260000
*                                     01270000
*                                     01280000
* LTORG Literal pool 01290000
*                                     01300000
ARG1 EQU B,B                First argument 01310000
REGEQU EQU B,B              01320000
END
    
```

Function Packages ...



```

TITLE 'USERFN REXX Function Package Glue Code' 00010000
LCLC &PACKAGE,&CR(1),&FUN(1) 00020000
***** 00030000
*                                     00040000
* Describe your function package here 00050000
*                                     00060000
*                                     00070000
*                                     00080000
&PACKAGE SETC 'USERFN' Name of the package 00090000
*                                     00100000
&CR(1) SETC ' ' Copyright notice 00110000
*                                     00120000
&FUN(1) SETC 'USER1' Name of function 00130000
&FUN(2) SETC 'USER2' Name of function 00140000
&FUN(3) SETC 'USER3' Name of function 00150000
*                                     00160000
*****EJECT 00170000
*****Start of Specifications***** 00180000
*                                     00190000
* This code is provided on an as-is basis. 00200000
*                                     00210000
*                                     00220000
*                                     00230000
* Module name: USERFN 00240000
*                                     00250000
* Descriptive name: Glue code for REXX function package 00260000
*                                     00270000
* Function: 00280000
*                                     00290000
* The following code resides in free storage and is capable 00300000
* of replying to LOAD and RESET. 00310000
* A LOAD call results in identifying the function (whose name 00320000
* is passed as parameter following LOAD) as entry point. 00330000
* A LOAD call without function specified will identify all 00340000
* functions in the function package as entry points. 00350000
* A RESET call from NUCXDROP will turn the functions off. 00360000
* A PURGE service call is ignored. 00370000
* To generate 00380000
* HASH USERFN 00390000
* LOAD USERFN 00400000
* GENM USERFN ( NOHAP 00410000
* Note: To get a MODULE which can execute on CMS release 5 and 00420000
* later -> generate the module on a CMS release > 5. 00430000
* Note: When user functions are compiled REXX with the DLINK 00440000
* compiler option, then you must explicitly specify 00450000
* the functions to be DLINKed explicitly on the LOAD 00460000
* command, and you must use the RLOSAVE option for the 00470000
* LOAD command. 00480000
* Example: 00490000
* USER1, USER2, and USER3 are functions in the package. 00500000
    
```

Function Packages ...



```

* USER1 contains a call to USER3 and a call to USERX. 00510000
* - compile USER1 with the DLINK compiler option 00520000
* - LOAD USERFN USERX ( RLOSAVE 00530000
* - GENM USERFN ( NOHAP 00540000
* If you do this on CMS rel > 5 you will get a module 00550000
* runnable on CMS rel 5 and later, which will reside 00560000
* above the 16 MB line if possible. 00570000
* USER1, USER2, and USER3 are the known functions, 00580000
* calls in USER1 to USER2 and USERX will use DLINK. 00590000
*                                     00600000
* Entry/exit conditions: 00610000
*                                     00620000
* Standard SVC conventions. 00630000
* RI points to a tokenized PLIST. 00640000
* This code can run with AMODE 24 or AMODE 31 00650000
* Exit: 00660000
* R15 = 0 - Function successfully loaded or exists 00670000
* - Functions unloaded 00680000
* - Service call, Endcmd call, ... 00690000
* R15-- 0 - Return code from unsuccessful NUCEXT LOAD 00700000
* - 1 LOAD or RESET, but no second argument 00710000
* or function not in package 00720000
* - 4 no first argument 00730000
*                                     00740000
*                                     00750000
* Maclibs: 00760000
* DMSPP or DMSGPJ 00770000
*                                     00780000
* Macros and control blocks: 00790000
* REGEQU 00800000
* CHSLFVFL 00810000
* NUCON 00820000
* Change Activity: 00830000
* 91-11-21 KH Added AMODE 31 capability 00840000
* 93-06-22 KH Added missing branch for service call with 00850000
* AMODE 24 (thanks to Roderic A. Davis for pointing 00860000
* out the problem and providing the solution). 00870000
* 93-06-28 KH Load all functions in response to a LOAD request 00880000
* without function name specified. 00890000
*                                     00900000
*****End of Specifications***** 00910000
*                                     SPACE 1 00920000
&PACKAGE RMODE ANY 00930000
&PACKAGE AMODE 31 00940000
*                                     SPACE 1 00950000
&PACKAGE CSECT, 00960000
USING NUCON,B Establish Addressability 00970000
USING *,R12 Establish Addressability 00980000
B STARTCOD Branch around header 00990000
DC CL8'&PACKAGE' Package ID 01000000
&MAX SETA N'&CR 00970000
    
```

Function Packages ...



```

&I SETA 1 00980000
.CRLOOP ANOP 00990000
AIF (&I GT &MAX).CRLOOPE 01000000
DC C'&CR(&I)' 01010000
&I SETA &I+1 01020000
AGO .CRLOOP 01030000
.CRLOOPE ANOP 01040000
* List of functions included in this pack, with their offsets 01050000
FUNNAME EQU 4,0 Offset & length of name 01060000
FUNOFFS EQU 0,4 Offset to the routine 01070000
LENTRY EQU 12 Length of a single entry 01080000
FUNLIST DS BF List of functions 01090000
&MAX SETA N'&FUN 01100000
&I SETA 1 01110000
.FULOOP ANOP 01120000
AIF (&I GT &MAX).FULOOPE 01130000
EXTRN &FUN(&I) Name of compiled program 01140000
DC A(&FUN(&I)-&PACKAGE) Offset of compiled prog 01150000
DC C'&RX' Name prefix 01160000
DC CL6'&FUN(&I)' Name 01170000
&I SETA &I+1 01180000
AGO .FULOOP 01190000
.FULOOP ANOP 01200000
EFUNLIST DC A("=) End fence 01210000
*                                     01220000
*                                     01230000
*                                     01240000
*                                     01250000
STARTCOD EQU * 01260000
LR R10,R14 Save return address 01270000
LR R11,R14 Save return address once more 01280000
CLC ARG1(0,R1),-CLB'LOAD' Is this a load? 01290000
BE CHKAARGS Yes, check for any arg's 01300000
BE DOOF Reset? Yes, turn off functions 01310000
SR R15,R15 In case of service call 01320000
L R2,'X'FF000000 Set HOB of register 01330000
LA R2,0(R2) LA will clear HOB in AMODE 24 01340000
LTR R2,R2 Br if AMODE 24 01350000
BZ PL2021 01360000
CLI 9C(R13),X'FE' Look at Call Type (XA) 01370000
B CONTAL 01380000
PL2021 CLM R1,0'1000',X'FE' Look at Call Type 01390000
CONTAL EQU * 01400000
BNLR R14 Return Service, Endcmd, ... 01410000
LA R15,4 Bad Plist, set error code 01420000
BR R14 .. and return 01430000
CHKAARGS EQU * 01440000
LA R15,1 Set possible return code 01450000
CLI ARG2(R1),X'FE' Any arguments passed? 01460000
BE ALLLOAD No, load all functions 01470000
*                                     01480000
    
```


Function Packages ...



```

* LOAD request. Check function name against FUNLIST.          01480000
* Only turn on the requested function.                        01490000
*                                                            01500000
PUSH USING          Save USING status                        01510000
USING DNUCX,R13    Use save area for PLIST                  01520000
AUTOLoad EQU      *                                        01530000
MVC DNLIST(LNLIST),MLIST Move skeleton to work area        01540000
L R3,R1           Save old plist pointer                   01550000
LA R4,LENTY      Length of FUNLIST entry                  01560000
LA R5,EFUNLIST   End of function table                    01570000
LA R2,FUNLIST    Start of function table                  01580000
LA R15,1         Set error return code                    01590000
CHECK1 EQU       *                                        01600000
CLC ARG2(R3),FUNLNAME(R2) Check against name              01610000
BE TURNON        Found - turn function on                 01620000
BXLE R2,R4,CHECK1 Loop for another check                  01630000
BR R10           Return with RC = 1                       01640000
*                                                            01640100
* LOAD request without function name, load all functions in  01640200
* package. Return with RC 0.                                01640300
*                                                            01640400
ALLLOAD EQU      *                                        01640500
LR R3,R1         Save old plist pointer                   01640600
LA R4,LENTY      Length of FUNLIST entry                  01640700
LA R5,EFUNLIST   End of function table                    01640800
SR R5,R4         last entry in function table             01640900
LA R2,FUNLIST    Start of function table                  01641000
NEXTFI EQU       *                                        01641100
MVC DNLIST(LNLIST),MLIST Move skeleton to work area        01641200
BAL R10,TURNON   Turn on the function                     01641300
BXLE R2,R4,NEXTFI Loop for another function               01641400
SR R15,R15      Set ok return code                        01641500
BR R11          Return                                    01641600
*                                                            01650000
TURNON EQU       *                                        01650000
MVC DNLNAME,FUNLNAME(R2) Copy startup name                 01650000
LA R1,DNLIST     -> PLIST                                  01650000
*                                                            01660000
* See if function is already a nucleus extension, make it   01690000
* a nucleus extension if not (CMS rel > 5)                  01700000
*                                                            01710000
LA R15,1         -1                                       01720000
LNR R15,R15     -1                                       01730000
ST R15,DNLADDR  Query form of NUCEXT plist                01740000
CLI CMSPROG,VMSPS Are we on CMS release 5                 01750000
BNH SV2021      Br if yes, use SVC 202                    01760000
L R15,'X'00000000 tok Plist, COPY/FENCE flags            01770000
SVC 204         tok Plist, COPY/FENCE flags               01780000
L R15,R15       Exists?                                    01790000
BZR R10        Yes, immediate return                      01800000
L R6,FUNOFFS(R2) Load address offset                     01810000
LA R6,0(R6,R12) True start address                       01820000

```

05/94 KH News from the REXX Compiler - Supplement

Function Packages ...



```

ST R6,DNLADDR    Add to startup PSW                       01830000
L R15,'X'00000000 tok Plist, COPY/FENCE flags           01840000
SVC 204         tok Plist, COPY/FENCE flags               01850000
BR R10          Return                                    01860000
*                                                            01870000
* See if function is already a nucleus extension, make it   01880000
* a nucleus extension if not (CMS rel 5)                  01890000
*                                                            01900000
SV2021 SVC 202    -Fall through if error                 01920000
DC AL4(1)       Exists?                                   01930000
LTR R15,R15     Yes, immediate return                     01940000
L R6,FUNOFFS(R2) Load address offset                     01950000
LA R6,0(R6,R12) True start address                       01960000
ST R6,DNLADDR   Add to startup PSW                       01970000
SVC 202        Br if yes, use SVC 202                    01980000
DC AL4(1)       Ignore errors                             01990000
BR R10          Return                                    02000000
POP USING       Restore USING status                      02010000
*                                                            02020000
* RESET request: switch off functions                      02030000
*                                                            02040000
DOOFF EQU       *                                        02050000
PUSH USING      Save USING status                        02060000
USING DNUCX,R13 Use save area for PLIST                  02070000
MVC DNLIST(LNLIST),MLIST Move skeleton to work area        02080000
L R1,DNLIST     -> to list                                 02090000
CLI CMSPROG,VMSPS Are we on CMS release 5                 02100000
BNH SV2022      Br if yes, use SVC 202                    02120000
SV2042 EQU      *                                        02130000
L R15,FUNOFFS(R5) Any more to cancel?                    02140000
LTR R15,R15     0 = all done ... Get out                  02150000
BZR R10        0 = all done ... Get out                  02160000
MVC DNLNAME(R5),FUNLNAME(R5) Copy startup name           02170000
L R15,'X'00000000 tok Plist, COPY/FENCE flags           02180000
SVC 204         tok Plist, COPY/FENCE flags               02190000
* (We ignore errors e.g.: function already cancelled)     02200000
LA R5,LENTY(R5) -> next item in FUNLIST                  02210000
B SV2042        -> next item in FUNLIST                  02220000
*                                                            02230000
SV2022 EQU      *                                        02240000
L R15,FUNOFFS(R5) Any more to cancel?                    02250000
LTR R15,R15     0 = all done ... Get out                  02260000
BZR R10        0 = all done ... Get out                  02270000
MVC DNLNAME(R5),FUNLNAME(R5) Copy startup name           02280000
SVC 202        tok Plist, COPY/FENCE flags               02290000
DC AL4(1)       Ignore errors                             02300000
LA R5,LENTY(R5) -> next item in FUNLIST                  02310000
B SV2022        -> next item in FUNLIST                  02320000
POP USING       Restore USING status                      02340000

```

05/94 KH News from the REXX Compiler - Supplement

Function Packages ...



```

* EJECT 02350000
* Equates 02360000
* 02370000
* 02380000
ARG1 EQU 0,0      First argument 02390000
ARG2 EQU 16,0    Second argument 02400000
L REGEQU         02410000
L CHSLEVEL      02420000
* 02430000
* PLIST for invoking 'NUCEXT' (setup as CANCEL PLIST) 02440000
* 02450000
NLIST DS 00      NUCEXT Plist 02460000
DC CL0'NUCEXT'  Name 02470000
NLNAME DC CL0'APACKAGE' Function name 02480000
DC X'FF'       System mask enabled 02490000
NLKEY DC X'04'  System key 02500000
NLFLAG DC AL1(SYSTEM) NUCEXT flag 02510000
DC X'00'       Spare flags 02520000
NLADDR DC A(0)  Entry point address 02530000
DC AL4(---)    private 02540000
NLSTART DC A(0) Start address 02550000
NLLEN DC F'0'  Length 02560000
LNLIST EQU *-NLIST length of list 02570000
* 02580000
* NUCEXT PLIST Flags: 02590000
* 02600000
SYSTEM EQU X'00' 02610000
* 02620000
* DSECT for NUCEXT plist 02630000
* 02640000
DNUCX DSECT      Based on register 13 02650000
DNLIST DS CL0 'NUCEXT' Name 02660000
DNLNAME DS CL0 'APACKAGE' Function name 02670000
DNLMASK DS X'FF'  Mask 02680000
DNLKEY DS X'04'  System Key 02690000
DNLFLAG DS AL1(SYSTEM) NUCEXT flag 02700000
DS X'00'       Spare flags 02710000
DNLADDR DS A     Entry point address (0=cancel) 02720000
DS AL4(---)    private 02730000
DLSTART DS A     Start address 02740000
DNLLEN DS AL4(FREELEN) Length 02750000
NUCON          02760000
END            02770000

```

05/94 KH News from the REXX Compiler - Supplement