Stages Built Into
NetRexx Pipelines 4.02
&
CMS Pipelines V7R1
and Their Differences

How to Read Syntax Diagrams

Special diagrams (often called railroad tracks) are used to show the syntax of external interfaces.

To read a syntax diagram, follow the path of the line. Read from left to right and top to bottom.

- The ▶▶--- symbol indicates the beginning of the syntax diagram.
- The ---▶ symbol, at the end of a line, indicates that the syntax diagram is continued on the next line.
- The ▶--- symbol, at the beginning of a line, indicates that the syntax diagram is continued from the previous line.
- The ---▶◀ symbol indicates the end of the syntax diagram.

Within the syntax diagram, items on the line are required, items below the line are optional, and items above the line are defaults.

Pipelines Builtin Stages

Show Stages Implemented in:

NetRexx Pipelines: ☑   **CMS Pipelines:** ☑

Show All Details: ☑ (Double click on a row to turn it on/off.)

Highlight NetRexx Only / CMS Only: ☑

> **diskw** **filew** **3.09**	**Replace or Create a File** ▶▶-->--*string*--▶◀ - delegates to **diskw**.
>> **diska** **filea** **3.09**	**Append to or Create a File** ▶▶-->>--*string*--▶◀ - delegates to **diska**.
>>mdsk	**Append to or Create a CMS File on a Mode** - Not implemented in Netrexx Pipelines.
>>mvs	**Append to a Physical Sequential Data Set** - Not implemented in Netrexx Pipelines.
>>oe	**Append to or Create an OpenExtensions Text File** - Not implemented in Netrexx Pipelines.
>>sfs	**Append to or Create an SFS File** - Not implemented in Netrexx Pipelines.
>>sfsslow	**Append to or Create an SFS File** - Not implemented in Netrexx Pipelines.
>mdsk	**Replace or Create a CMS File on a Mode** - Not implemented in Netrexx Pipelines.

>mvs	**Rewrite a Physical Sequential Data Set or a Member of a Partitioned Data Set** • Not implemented in Netrexx Pipelines.
>oe	**Replace or Create an OpenExtensions Text File** • Not implemented in Netrexx Pipelines.
>sfs	**Replace or Create an SFS File** • Not implemented in Netrexx Pipelines.
< *diskr* *disk* *file* *filer* *3.09*	**Read a File** ▶▶--<--*string*--▶◀ • Implemented as in CMS; delegates to **diskr**.
<mdsk	**Read a CMS File from a Mode** • Not implemented in Netrexx Pipelines.
<mys	**Read a Physical Sequential Data Set or a Member of a Partitioned Data Set** • Not implemented in Netrexx Pipelines.
<oe	**Read an OpenExtensions Text File** • Not implemented in Netrexx Pipelines.
<sfs	**Read an SFS File** • Not implemented in Netrexx Pipelines.
<sfsslow	**Read an SFS File** • Not implemented in Netrexx Pipelines.
-- *comment* *3.09*	**Comment Stage, No Operation** ┌**NetRexx**┐ ▶▶--+- -- ----+--+-------+------▶◀ +-COMMENT-+ +-*string*-+ • delegates to **comment**. • Not in CMS Pipelines; • This is a STAGE, not a programming comment. It must have a SPACE after **--**. • It must have either a stageEnd or pipeEnd. • If used before a driver stage, it must have a pipeEnd.
3277bfra	**Convert a 3270 Buffer Address Between Representations** • Not implemented in Netrexx Pipelines.
3277enc	**Write the 3277 6-bit Encoding Vector** • Not implemented in Netrexx Pipelines.

64decode
decode64
3.11
	Decode Base-64 Format
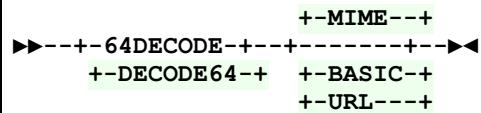	
	``
	                            +-MIME--+
	►►--+-64DECODE-+--+-------+--►◄
	    +-DECODE64-+ +-BASIC-+
	                  +-URL---+
	``
	
	• NOTE: CMS is only 64DECODE, and does not have the options; it does MIME.
	• BASIC - Output is mapped to a set of characters lying in A-Za-z0-9+/. The encoder does not add any line feed in ou
	  the decoder rejects any character other than A-Za-z0-9+/.
	• URL - Output is mapped to set of characters lying in A-Za-z0-9+_. Output is URL and filename safe.
	• MIME - Output is mapped to MIME friendly format. Output is represented in lines of no more than 76 characters ea
	  uses a carriage return '\r' followed by a linefeed '\n' as the line separator. No line separator is present to the end of t
	  encoded output.
	• 3.11: New to NetRexx. Add MIME, BASIC, & URL options.

64encode
encode64
3.11
	Encode to Base-64 Format
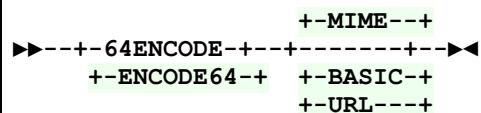	
	``
	                            +-MIME--+
	►►--+-64ENCODE-+--+-------+--►◄
	    +-ENCODE64-+ +-BASIC-+
	                  +-URL---+
	``
	
	• NOTE: CMS is only 64DECODE, and does not have the options; it does MIME.
	• BASIC - Output is mapped to a set of characters lying in A-Za-z0-9+/. The encoder does not add any line feed in ou
	  the decoder rejects any character other than A-Za-z0-9+/.
	• URL - Output is mapped to set of characters lying in A-Za-z0-9+_. Output is URL and filename safe.
	• MIME - Output is mapped to MIME friendly format. Output is represented in lines of no more than 76 characters ea
	  uses a carriage return '\r' followed by a linefeed '\n' as the line separator. No line separator is present to the end of t
	  encoded output.
	• 3.11: New to NetRexx. Add MIME, BASIC, & URL options.

? *help*	**Display Help for Pipelines** • Not implemented in Netrexx Pipelines.

abbreviation
abbreviatio
abbreviati
abbreviat
abbrevi
abbrev
	Select Records that Contain an Abbreviation of a Word in the First Positions
	
	``
	►►--ABBREViation-(1)-+----------------------------------+--►◄
	                     +-word-+-------------------------+-+
	                            +-number-+-------------+-+
	                                     +-ANYcase-(2)-+
	                                     +-CASEANY-----+
	                                     +-CASEIGNORE--+
	                                     +-IGNORECASE--+
	                                     +-CASELESS----+
	``
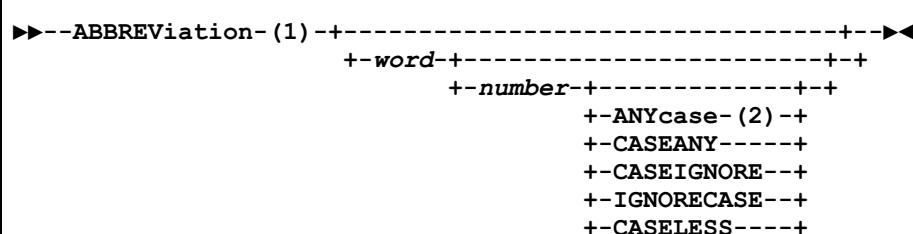	
	• (1) ABBREViation must be ABBREV in CMS
	• (2) ANYcase must be ANYCASE in CMS

acigroup	**Write ACI Group for Users** • Not implemented in Netrexx Pipelines.
addrdw	**Prefix Record Descriptor Word to Records** • Not implemented in Netrexx Pipelines.

adrspace	**Manage Address Spaces** • Not implemented in Netrexx Pipelines.
aftfst	**Write Information about Open Files** • Not implemented in Netrexx Pipelines.
aggrc	**Compute Aggregate Return Code** `▶▶--AGGRC--▶◀`
all	**Select Lines Containing Strings (or Not)** • Not implemented in Netrexx Pipelines.
alserv	**Manage the Virtual Machine's Access List** • Not implemented in Netrexx Pipelines.
apldecode	**Process Graphic Escape Sequences, Old APL language** • Not implemented in Netrexx Pipelines.
aplencode	**Generate Graphic Escape Sequences, Old APL language** • Not implemented in Netrexx Pipelines.
append	**Put Output from a Device Driver after Data on the Primary Input Stream** `▶▶--APPEND--string------▶◀`
array	**Read or Write an Array** • Pipes for NetRexx
arraya	**Read or Write an Array** • Pipes for NetRexx
arrayr	**Read or Write an Array** • Pipes for NetRexx
arrayw	**Read or Write an Array** • Pipes for NetRexx
asatomc	**Convert ASA Carriage Control to CCW Operation Codes. Old printer control** • Not implemented in Netrexx Pipelines.
asmcont	**Join Multiline Assembler Statements** • Not implemented in Netrexx Pipelines.
asmfind	**Select Statements from an Assembler File as XEDIT Find** • Not implemented in Netrexx Pipelines.
asmnfind	**Select Statements from an Assembler File as XEDIT NFind** • Not implemented in Netrexx Pipelines.
asmxpnd	**Expand Joined Assembler Statements** • Not implemented in Netrexx Pipelines.
beat	**Mark when Records Do not Arrive within Interval** • Not implemented in Netrexx Pipelines.

between
3.09
	Select Records Between Labels
	
	``
	►►--BETWEEN--| case |-+-delimitedString1-+-+-number----------+---►◄
	                        +-Xhexstring1------+ +-delimitedString2-+
	                        +-Hhexstring1------+ +-Xhexstring2------+
	                        +-Bbinstring1------+ +-Hhexstring2------+
	                                            +-Bbinstring2------+
	
	case:
	|--+------------+--|
	   +-ANYcase-----+
	   +-CASEANY-----+
	   +-CASEIGNORE--+
	   +-IGNORECASE--+
	   +-CASELESS----+
	``

block	**Block to an External Format** • Not implemented in Netrexx Pipelines.
browse	**Display Data on a 3270 Terminal** • Not implemented in Netrexx Pipelines.

buffer	**Buffer Records**
	
	``
	►►--BUFFER--+--------------------------+-------►◄
	            +-number--+----------------+-+
	                      +-delimitedString-+
	``

c14to38	**Combine Overstruck Characters to Single Code Point. Old printer.**

casei	**Run Selection Stage in Case Insensitive Manner**
	
	``
	►►--CASEI--+-------------------+----+---------+-----►
	           +-ZONE--|inputRange |-+(1) +-REVERSE-+(1)
	
	►--stage-+--------+--►◄
	         +-string-+
	``
	
	• (1) CMS Pipelines only.

change	**Substitute Contents of Records**

```
►►--CHANGE--+-------------+--+----------------+-->
            +-| anycase |-+  +-inputRange------+
                             |   ⌐-------+      |
                             +-(----range-+--)-+

      +-FROMTO-(1)+
►--+--+------------+-| changeString |-----------------------+-->
   | +-FROM-(1)+                     +-TO-(1)+              |
   +-+---------+-delimitedString-+-------+-delimitedString-+

►---+----------------------+---►◄
    | +-FOR-(1)+            |
    +-+--------+-numorstar-+


  changeString:
  |--delimiter--string--delimiter--string--delimiter--|

  anycase:
     +-RESPECTCASE-(1)+
  |--+----------------+--|
     +-ANYcase--------+
     +-CASEANY--------+
     +-CASEIGNORE-----+
     +-IGNORECASE-----+
     +-CASSLESS-------+
```

- (1) NetRexx Pipelines only.

changeparse
changepars
changepar
changepa
changep
3.11
	Change the contents of records, using Rexx Parse. Calculations can be done.
	
	``
	┌NetRexx───┐
	│ +-FROM-+ +-TO-+ │
	│ ►►--CHANGEParse--+------+-parse_template_Dstring-+----+-output_template_Dstring--► │
	│ │
	│ ►--+----------------------+--► │
	│ +-BY-by_NetRexx_Dstring-+ │
	│ │
	│ ►--+----------------------------+--+--------------------------------+--►◄ │
	│ +-FIRST-first_NetRexx_Dstring-+ +-FINALLY-finally_template_Dstring-+ │
	└──┘
	``
	
	- Records are parsed via the parse_template_delimited_string.
	- Variables are named $n, where n is 1 to 9.
	- The by_NetRexx_delimited_string is interpreted. This is 0 or more semicolon separated NetRexx statements, proba... the $n variables, which can have the value altered.
	- Other variables may be used, and are persistent while the stage is active, so can be used as accumulators.
	- The values of the variables are put into the output_template_delimited_string replacing $n.
	- For a literal $n that won't be changed, use $$n.
	- A first_NetRexx_delimited_string, if present, is interpreted before reading any record from the primary input steam. ... or more semicolon separated NetRexx statements, probably using the $n variables. Any variables used in the by_NetRexx_delimited string must be defined here.
	- A finally_template_delimited_string, if present, is written as a final output record after the primary input stream is fini... using the $n's.
	- Any keyword phrases must, in any order, follow any non-keyworded FROM & TO phrases.
	- This is NetRexx Pipelines only, not in CMS.
	
	Examples:
	
	- **changeparse / 2 $1 +1/ /The second letter is "$1". $$1 won't be changed./**
	- **changeparse from / 2 $1 +1/ to /The second letter is "$1". $$1 won't be changed./**
	- **changeparse from / . $2 . 50 $5 +5 / to /The product is $1/ by /$1 = $2 * $5/**
	- **changeparse from / . $2 . 50 $5 +5 / ,**
	  to /The product is $1/ ,
	  by /$1 = $2 * $5;$3 = $3 + $1/ ,
	  first /$3 = 0/ ,
	  finally /$3 is the total/

changeregex *changerege* *changereg* *changere* *changer* *3.09*	**Substitute Contents of Records using Java Regular Expressions** ``` ┌NetRexx───┐ │ +-FROM-+ +-TO-+ +-ONE-+ │ │ ►►--CHANGERegex--+------+-delimitedString-(1)-+----+-delimitedString-(2)-+-----+--►◄ │ │ +-ALL-+ │ └──┘ ``` - Uses the Java RegEx classes and its dialect of RegEx. See Java's **Pattern** class and **replaceFirst** and **replaceAll** ... of **String** for full documentation. - (1) First, FROM, delimitedString is a Java *RegEx expresion* for what is to be replaced. - (2) Second, TO, delimitedString is the replacement string. It may contain elements from the first one. - Ths is NetRexx Pipelines only, not in CMS.

chop
truncate
truncat
trunca
trunc
	Truncate the Record
	
	``
	                       +--80----------------+
	 ►►--+--CHOP----+--+--------------------+--►◄
	     +-TRUNCate-+ +-snumber------------|
	                    +--| stringtarget |--+
	 stringtarget:
	  |-+-------------+-+---------------------+-+-----+-| target |-|
	    +-ANYcase-----+
		+-BEFORE-+	+-NOT-+
			    +-CASEANY-----+ +-+---------+-+--------+-+
			    +-CASEIGNORE--+ +-snumber-+ +-AFTER--+
			    +-IGNORECASE--+
			    +-CASELESS----+
			
			 target:
			  |--+--xrange---------------------+--|
			     +--+-STRing--+--delimitedString-+
			         +--ANYof--+
			``

cipher	**Encrypt and Decrypt Using a Block Cipher** - Not implemented in Netrexx Pipelines.

ckddeblock	**Deblock Track Data Record** • Not implemented in Netrexx Pipelines.
cmd **command**	**Issue OS Commands, Write Response to Pipeline** ``` ►►--+-CMD-----+--+----------+------------►◄ +-COMMAND-+ +--string--+ ``` • input stream 0 is for commands • input stream 1 is stdin • output stream 0 is stdout • output stream 1 is the return code • output stream 2 is stderr
cms	**Issue CMS Commands, Write Response to Pipeline** '

collate
3.11
	Collate Streams
	
	``
	                                +-NOPAD----+
	►►-COLLATE--+-----------+-+----------+-+-------------+--►
	            +-STOP ANYOF-+ +-PAD-xorc-+ +-| anycase |-+
	
	                                        +-MASTER DETAIL---+
	  ►--+--------------------------+--+-----------------+--►
	     +-inputRange-+-----------+-+ +-MASTER----------+
	                  +-inputRange-+ +-DETAIL-+--------+
	                                            +-MASTER-+
	
	  ►--+------------------------------+--►◄
	     +-SEParator-delimitedString-(1)(2)-+
	
	  anycase:
	     +-RESPECTCASE-(1)+
	  |--+---------------+--|
	     +-ANYcase--------+
	     +-CASEANY--------+
	     +-CASEIGNORE-----+
	     +-IGNORECASE-----+
	     +-CASSLESS-------+
	``
	
	• Not implemented in Netrexx Pipelines.
	
	• (1) NetRexx Pipelines only.
	• (2) delimitedString record is put before each Master Record
	  (or after if DETAIL MASTER order) on the primary output stream.
	• 3.11 New to NetRexx Pipelines. Add SEParator option.

combine	**Combine Data from a Run of Records**
	
	``
	►►--COMBINE--+----------------------+--+-Or---------+--►◄
	
		+-1----------------+	+-aNd---------+
			             +-+------------------+--+ +-AND---------+
			
				+-number-----------+	+-eXclusiveor-+
					
						+-*----------------+	+-EXClusiveor-+
							
								+-KEYLENgth--number-+	+-FIRST-(2)---+
									
										+-ALLEOF-(1)+	+-LAST-(2)----+
											             +-STOP-(1)-+ +-+
											                        +-ANYEOF-(1)+
											``
											
											• (1) Only for use with secondary input streams. Only options from this column usable with any secondary input strea
											  (This is poorly documented in CMS Pipelines. This is a best guess of their intentions.)
											• (2) Not usable with STOP and secondary streams.

command cmd	**Issue OS Commands, Write Response to Pipeline**
| | ```
►►--+-COMMAND-+--+---------+------------►◄
     +-CMD-----+  +--string--+
``` |
| | - input stream 0 is for commands
- input stream 1 is stdin
- output stream 0 is stdout
- output stream 1 is the return code
- output stream 2 is stderr |
| comment
--
3.09 | **Comment stage** |
| | ```
┌NetRexx─────────────────────────────┐
│►►--+--COMMENT--+--+---------+-----►◄ │
│    + -- -------+  +--string--+      │
└────────────────────────────────────┘
``` |
| | - Not in CMS Pipelines;
- This is a STAGE, not a programming comment. It must have a SPACE after --.
- It must have either a stageEnd or pipeEnd.
- If ended with a stageEnd, it passes records through on primary input to output streams.
- If ended with a pipeEnd, it does NOT pass records through.
- If used before a driver stage, it must have a pipeEnd. |
| compare | **Compare Primary and Secondary Streams, Write the Result** |
| | ```
┌NetRexx──────────────────────────────────────────────────────────────────┐
│                +-TRINARY-+  (1)                       +-PAD SPACE-+       │
│►►--COMPARE--+---------+-------------------+---+----------+--+------+--►◄   │
│                +-BINARY--+  (2)            |    +-PAD-xorc--+  +-ECHO-+    │
│                |                           |    |                         │
│                | ┌-------------------------+   |                         │
│                +--ANY delimitedString------+--+   (4) (5)                 │
│                +--EQUAL delimitedString----+      (4)                     │
│                +--LESS delimitedString-----+   (3) (4)                    │
│                +--MORE delimitedString-----+   (3) (4)                    │
│                +--NOTEQUAL delimitedString-+      (4)                     │
└──────────────────────────────────────────────────────────────────────────┘
``` |
	- (1) -1 = Primary is shorter/less, 0 = equal, 1 = Secondary is shorter/less
	- (2) 0 = equal, 1 = not equal
	- (3) Primary is LESS/shorter (or MORE/longer) than secondary
	- (4) *DStrings* can use any of the following escapes (or the lowercase) for the unequal situation:
	  ○ \C (count) for the record number,
	  ○ \B (byte) for column number
	  ○ \P (primary) for the primary stream record
	  ○ \S (secondary) for the secondary stream record
	  ○ \L (Least) for the stream number that is shorter, -1 if equal
	  ○ \M (Most) for the stream number that is longer, -1 if equal
	- (5) Equal or not, this DString precedes any of the others.
	- (6) This is NetRexx Pipelines only, not included in CMS
	- (7) In reporting \P & \S, control charactors, except new line, \n, are transliterated to [blob, 219.d2c()]
	- (8) Without ECHO, this stops and reports at first non-compare. With ECHO, each primary input is reported; after firs
	  compare primary input stream records continue to be read and reported, but no testing is done.
	- (9) Options work in any order
	- Input streams:
	  ○ 0: Data 1
	  ○ 1: Data 2
	- Output streams:
	  ○ 0: Result (single record, possibly multiple lines)
	  ○ 1: Last primary record read at first no match, or end of stream
	  ○ 2: Last secondary record read at first no match, or end of stream

configure	**Set and Query CMS Pipelines Confguration Variables**
	- Not implemented in Netrexx Pipelines.

console
consol
conso
cons
cons
terminal
termina
termin
termi
term
3.11
	Read or Write the Terminal in Line Mode
	
	``
	►►--+-CONSole--+--+--------------------+--+----------------------------+--| CMSonly
	     +-TERMinal-+ +-EOF--delimitedString-+ +-PRfix--+-delimitedString-(1)+
	                    +-NOEOF---------------+ +-PRompt-+
	
	  CMSonly:
	    -|-+---------------+-(2)-|-
	       +-DIRECT---------+
	       +-ASYNchronously-+
	       +-DARK-----------+
	``
	
	- (1) NetRexx only
	  On first stage, delimitedString is put out as a prompt
	  On other stages, each line is prefixed with delimitedString
	  Outout to next stage does NOT include delimitedString
	  Either keyword can be used for either stage
	- (2) CMS only

copy	**Copy Records, Allowing for a One Record Delay** ``` ►►--COPY---------►◄ ```

count	**Count Lines, Blank-delimited Words, and Bytes**
	
	``
	                 ┌--------------------+
	►►--COUNT--+-+-+-CHARACTErs-+-+-+------►◄
	
		+-CHARS------+ |
		
			+-BYTES------+ |
			           +-WORDS----------+
			           +-+-LINES---+----+
			
				+-RECORDS-+ |
				           +-MINline--------+
				           +-MAXline--------+
				``

cp	**Issue CP Commands, Write Response to Pipeline** - Not implemented in Netrexx Pipelines.
crc	**Compute Cyclic Redundancy Code** - Not implemented in Netrexx Pipelines.
dam	**Pass Records Once Primed** ``` ►►--DAM------------►◄ ```
dateconvert *dateconver* *dateconve* *dateconv* *3.09*	**Convert Date Formats**

```
►►--DATECONVert--+---------------+-------►
                +--inputRange-(3)-+

     +-SHOrtdate ISOdate-----------------+ +-WINDOW -50---------+
  ►-+----------------------------------+-+--------------------+--►
    |                      +-ISOdate---------+ +-WINDOW-signednumber-+
    +-| Inputformat |-+-----------------+ +-BASEYEAR-yearnumber-+
    |      +-PREFACE-+ | +-| Outputformat |-+
    +-NOW-+-(5)-----+-+
            +-APPEND--+


                     +--MIDNIGHT--(4)-+
  ►--+----------+--+---------------+----------►◄
     +--TIMEOUT--+   +--NOON--(4)-----+
```

```
Inputformat,       Outputformat:
SHOrtdate      }   mm/dd/yy hh:mm:ss.uuuuuu
USA_SHORT      }
REXX_DATE_U    }

FULldate       }   mm/dd/yyyyyyy hh:mm:ss.uuuuuu
USA            }

ISO_SHORT          yy-mm-dd hh:mm:ss.uuuuuu
ISOdate            yyyyyyy-mm-dd hh:mm:ss.uuuuuu
DB2_SHORT          yy-mm-dd-hh.mm.ss.uuuuuu
DB2                yyyyyyy-mm-dd-hh.mm.ss.uuuuuu
VMDATE   (2)
NORMAL             dd mmm yyyyyyy hh:mm:ss.uuuuuu
CSL_SHORT      }   yy/mm/dd hh:mm:ss.uuuuuu
REXX_DATE_O    }
CSL                yyyyyyy/mm/dd hh:mm:ss.uuuuuu
PIPE_SHORT         yymmddhhmmssuuuuuu
PIPE           }   yyyymmddhhmmssuuuuuu
REXX_DATE_S    }
EURSHORT           dd.mm.yy hh:mm:ss.uuuuuu
EUR                dd.mm.yyyyyyy hh:mm:ss.uuuuuu
JULIAN_SHORT       yy.ddd hh:mm:ss.uuuuuu
JULIAN             yyyyyyy.ddd hh:mm:ss.uuuuuu
TOD_ABSOLUTE }   (2)
TODABS       }   (2)
SCIENTIFIC_ABSOLUTE }   (2)
SCIABS              }   (2)
POSIX              ssssssss
TOD_RELATIVE }   (2)
TODREL       }   (2)
SCIENTIFIC_RELATIVE }   (2)
SCIREL              }   (2)
MET   (2)


The following can be REXX_DATE_x, REXXx, or Rx
REXX_DATE_B   (2)
REXX_DATE_C   (2)
REXX_DATE_D        ddd hh:mm:ss.uuuuuu
REXX_DATE_E        dd/mm/yy hh:mm:ss.uuuuuu
REXX_DATE_E_LONG   dd/mm/yyyyyyy hh:mm:ss.uuuuuu
REXX_DATE_J        yyddd hh:mm:ss.uuuuuu
REXX_DATE_J_LONG   yyyyddd hh:mm:ss.uuuuuu
REXX_DATE_M        mmmmmmmmm (output only)
REXX_DATE_N_SHORT  dd mmm yy hh:mm:ss.uuuuuu
REXX_DATE_N        dd mmm yyyy hh:mm:ss.uuuuuu
REXX_DATE_W        wwwwwwww (output only)
```

- (1): SPACE is optional here.
- (2) Not implemented in NetRexx Pipelines at this time; mainly mainframe useful only.
- (3): NetRexx Pipelines uses IRange which gives a superset of range options.
- (4): NetRexx Pipelines only. What time to assume if blank time on input.
- (5): NetRexx Pipelines only.
  - Use current local date time.
  - Any Inputrange is ignored.
  - Any output format can be used.
  - PREFACE Write the date record before passing the input to the output.

deal	**Pass Input Records to Output Streams Round Robin**

```
                       +-STOP--ALLEOF-------------------+
►►--DEAL--+------------------------------------------+------►◄
                       +-STOP--+-ALLEOF-(2)+-------------+
                       |       +-ANYEOF----+             |
                       |       +-number----+             |
                       +-SECONDARY--+-----------+--------+
                       |            +-RELEASE---+        |
                       |            +-LATCH-(1)-+        |
                       +-KEY--inputRange--+-------+------+
                       |                  +-STRIP-+      |
                       +-STREAMid--inputRange--+-------+-+
                                               +-STRIP-+
```

- (1) Not yet in NetRexx Pipelines
- (2) Not CMS
- Since Java dispatches the stage threads, DEAL may not see a sever immediately, as the severing thread can get multitasked. This can make options like 'ANYEOF' work in unexpected ways.

deblock	**Deblock External Data Formats**

```
                         +-80------+
              +--FIXED--+-number--+--+-------------+-----------+
              |                      +--PAD--xorc--+ (1)       |
►►--DEBLOCK--+--------------------------------------------------+-►◄
              +-+-C----------------------+-+-----------+-+-----+
                +-J----------------------+ +-TERMINATE-+ +-EOF-+
                +-CRLF-------------------+
                +-LINEND xorc------------+
                +-STRING--delimitedString-+
```

- CMS has many more mainframe centric formats that NetRexx Pipelines does not process.
- (1) Not CMS Pipelines

decode64 *64decode* *3.11*	**Decode Base-64 Format**

```
-NetRexx-
                        +-MIME--+
►►--+-64DECODE-+--+-------+--►◄
    +-DECODE64-+  +-BASIC-+
                  +-URL---+
```

- NOTE: CMS is only 64DECODE, and does not have the options; it does MIME.
- BASIC - Output is mapped to a set of characters lying in A-Za-z0-9+/. The encoder does not add any line feed in ou the decoder rejects any character other than A-Za-z0-9+/.
- URL - Output is mapped to set of characters lying in A-Za-z0-9+_. Output is URL and filename safe.
- MIME - Output is mapped to MIME friendly format. Output is represented in lines of no more than 76 characters ea uses a carriage return '\r' followed by a linefeed '\n' as the line separator. No line separator is present to the end of t encoded output.
- 3.11: New to NetRexx. Add MIME, BASIC, & URL options.

delay	**Suspend Stream**

- Not implemented in Netrexx Pipelines.

devinfo	**Write Device Information**

- Not implemented in Netrexx Pipelines.

dfsort	**Interface to DFSORT/CMS**

- Not implemented in Netrexx Pipelines.

diage4	**Submit Diagnose E4 Requests**

- Not implemented in Netrexx Pipelines.

dict *hash*	**Read or Write a Dictionary**

- Pipes for NetRexx only.

dicta *hasha*	**Read or Write a Dictionary** • Pipes for NetRexx only.
dictr *hashr*	**Read or Write a Dictionary** • Pipes for NetRexx only.
dictw *hashw*	**Read or Write a Dictionary** • Pipes for NetRexx only.
digest	**Compute a Message Digest** • Not implemented in Netrexx Pipelines.
disk *diskr* *<* *file* *filer*	**Read a File** ┌**NetRexx**───── ▶▶--DISK--*string*--▶◀ • As in CMS, equivalent to diskr (Pipes for NetRexx Only) or **<**.
diska *>>* *filea*	**Append to or Create a File** ┌**NetRexx**───── ▶▶--DISKA--*string*--▶◀
diskback *fileback*	**Read a File Backwards** • Not implemented in Netrexx Pipelines.
diskfast *filefast*	**Read, Create, or Append to a File** • Not implemented in Netrexx Pipelines.
diskid	**Map CMS Reserved Minidisk** • Not implemented in Netrexx Pipelines.
diskr *disk* *<* *file* *filer*	**Read a File** ┌**NetRexx**───── ▶▶--DISKR--*string*--▶◀ • As in CMS, equivalent to diskr (Pipes for NetRexx Only) or **<**.
diskrandom *filerandom*	**Random Access a File** • Not implemented in Netrexx Pipelines.
diskslow *fileslow*	**Read, Create, or Append to a File** • Not implemented in Netrexx Pipelines.
diskupdate *fileupdate*	**Replace Records in a File** • Not implemented in Netrexx Pipelines.
diskw *filew* *>*	**Replace or Create a File** ▶▶-->--*string*--▶◀
display *3.11*	**Output to Web Browser**

```
┌─NetRexx─────────────────────────────────────────────────────┐
│ ►►──DISPLAY──+─────────────────────────────────────────+──►  │
│             | +-AS-+  +-TEXT--+                         |     │
│             +-+---+--+-------+--+----------------+            │
│                      +-PRE---+  +-OPTIONs-QString-+          │
│                      +-HTML--+                               │
│                      +-NONE--+                               │
│                      +-TABLE-+                               │
│                      +-NOTAG-+                               │
│                                                              │
│  ►--+---------------+--►                                      │
│     +-TITLE--QString-+                                        │
│                                                              │
│  ►--+---------------+--►                                      │
│     +-STYLE--QString-+                                        │
│                                                              │
│  ►--+-------------------------+--►                            │
│     |           +-.-+ +-html----+ |                          │
│     +-FILETYPE-+---+-+-QString-+-+                           │
│                                                              │
│  ►--+----------------------+--►◄                              │
│     |            +-PipeDisp--+ |                             │
│     +-FILENAME-+----------+-+                                │
│                +-QString---+                                 │
└──────────────────────────────────────────────────────────────┘
```

- DISPLAY works similar to and as a replacement for CONSOLE for output. But instead of going to the terminal windo goes to a HTML file browser tab. This allows for HTML+CSS tags to control fonts, colors, and layout. This stage ca found in examples-pipes.
- To work, these are required outside Pipelines and NetRexx:
  - A working HTML browser program
  - The operating system to associate the filetype "html" with the browser, so the Pipelines stage "COMMAND PipeDisp.html" does call the browser and display the file.
  - The system have a Temp directory, known to Java.
- The DISPLAY stage overwrites the named file, by default PipeDisp.html, in the system Temp directory, then calls th COMMAND stage to display it. The file is not erased automatically by this stage.
- Each DISPLAY stage invocation opens a new browser tab, which remains open.
- The AS option causes the data to be surrounded by html tags.
  - The default TEXT or PRE puts on <pre> and </pre>. Most browsers use:
    - Fixed width font
    - Display all the white spaces: line feeds and multiple spaces
  - HTML uses <html> and </html>. Most browsers use:
    - Variable width font
    - Consolidate strings of white space into a single space
    - All the HTML tags
  - TABLE uses <table> and </table>
    - Expects the data records to begin with <tr><td> (or <tr><th>)
  - NOTAG uses <pre> & </pre>, but first converts all & characters to the entity & and < characters to < so H tags are not processed.
  - NONE uses no extra tags. Most browsers use:
    - HTML display
- OPTIONs QString is included in the opening tag for the AS option. This could be CLASS, STYLE, or other options.
- TITLE QString adds <title>delimitedString</title> to the beginning of the output. This should show as the title in the tab.
  Note: This officially should go into a HEAD section; here it won't be there. Most modern browsers will honor it anypl file. If it is not honored as a tag, QString will be the top line of the display.
- STYLE QString adds <link rel="stylesheet" href="QString"> to the beginning of the output. This should include and u named stylesheet. The name may have relative path names, or be an absolute file name. If there are spaces, enclo quotes.
  Note: This officially should go into a HEAD section; here it won't be there. Most modern browsers will honor it anypl file. If it is not honored as a tag, it will not show -- except in the NOTAG option. The file itself is copied from its state into the system Temp directory, overwriting any existing file. This file is not erased automatically by this stage.
  QString: It is optional to enclose the name in quotes, but quotes are required if the name includes spaces.
- FILETYPE may be used to change the default "html". This permits use of other types that MAY be preprocessed if t system, external to Pipelines, is set up to recognize it, for example, "JSP" or "PHP". A "dot" is optional; only one wil
  Note: filetypes other than .html may be handled by the system by some program other than the browser.
  QString: It is optional to enclose the type in quotes.
- FILENAME may be used to write and display another file. It may include a path designation, either absolute or relati relative path is based on the working directory. If no path is specified in the name, the system Temp directory, as de by Java, is used.
  QString: It is optional to enclose the name in quotes, but quotes are required if the name includes spaces.
- Records from the primary input stream are also put out on the primary output stream unchanged, if it is connected.

drop	**Discard Records from the Beginning or the End of the File**
	
	``
	          +-FIRST-+ +-1----------+
	▶▶--DROP--+-------+--+------------+--+-------+------▶◀
	          +-LAST--+ +-snumber-(1)+ +-BYTES-+
	                     +-*----------+
	``
	
	  • (1) CMS: must be positive.
	     NetRexx Pipelines: negative reverses FIRST/LAST, so DROP FIRST -3 is the same as DROP LAST 3.

duplicate
duplicat
duplica
duplic
dupli
dupl
dup
	Copy Records
	
	``
	                    +-1------+
	▶▶--DUPlicate-(1)-+--------+--------▶◀
	                    +-number-+
	                    +-*------+
	                    +- -1----+
	``
	
	  • (1) CMS is DUPlicat due to 8-character name limitation

elastic	**Buffer Sufficient Records to Prevent Stall** ``` ▶▶--ELASTIC----------▶◀ ```

encode64
64encode
3.11
	Encode to Base-64 Format
	
	``
	┌NetRexx┐
	                        +-MIME--+
	▶▶--+-64ENCODE-+--+-------+--▶◀
	    +-ENCODE64-+ +-BASIC-+
	                   +-URL---+
	``
	
	  • NOTE: CMS is only 64DECODE, and does not have the options; it does MIME.
	  • BASIC - Output is mapped to a set of characters lying in A-Za-z0-9+/. The encoder does not add any line feed in ou
	    the decoder rejects any character other than A-Za-z0-9+/.
	  • URL - Output is mapped to set of characters lying in A-Za-z0-9+_. Output is URL and filename safe.
	  • MIME - Output is mapped to MIME friendly format. Output is represented in lines of no more than 76 characters eac
	    uses a carriage return '\r' followed by a linefeed '\n' as the line separator. No line separator is present to the end of t
	    encoded output.
	  • 3.11: New to NetRexx. Add MIME, BASIC, & URL options.

eofback	**Run an Output Device Driver and Propagate End-of-?le Backwards** • Not implemented in Netrexx Pipelines.
escape	**Insert Escape Characters in the Record** • Not implemented in Netrexx Pipelines.
fanin	**Concatenate Streams** ``` ▶▶--FANIN--+-------------+-------▶◀ 	⤶--------+ | +----stream-+-+ ```
faninany	**Copy Records from Whichever Input Stream Has One** ``` ▶▶--FANINANY----------▶◀ ```
fanintwo	**Pass Records to Primary Output Stream**

fanout	**Copy Records from the Primary Input Stream to All Output Streams**
	
	``
	                        +-STOP--ALLEOF-----------+
	►►--FANOUT--+------------------------+------->◄
	                        +-STOP--+-ANYEOF-------+-+
	                                +-ALLOF--(1)---+
	                                +-number-------+
	``
	
	• (1) CMS only

fanouttwo	**Copy Records from the Primary Input Stream to Both Output Streams**
fbaread	**Read Blocks from a Fixed Block Architecture Drive** • Not implemented in Netrexx Pipelines.
fbawrite	**Write Blocks to a Fixed Block Architecture Drive** • Not implemented in Netrexx Pipelines.
fblock	**Block Data, Spanning Input Records** ``` ►►--FBLOCK--number--+--------+---------------►◄ +--xorc--+ ```
file *filer* *disk* *diskr*	**Read a File** ┌NetRexx┐ ►►--FILE--*string*--►◄
filea *diska* *>>*	**Append to or Create a File** ►►--FILEA--*string*--►◄
fileback *diskback*	**Read a CMS file backwards** • Not implemented in Netrexx Pipelines.
filedescriptor	**Read or Write an OpenExtensions File that Is Already Open** • Not implemented in Netrexx Pipelines.
filefast *diskfast*	**Read or write a CMS file** • Not implemented in Netrexx Pipelines.
filer *file* *disk* *diskr* *<*	**Read a File** ┌NetRexx┐ ►►--FILER--*string*--►◄
filerandom *diskrandom*	**Read specific records from a CMS file**
fileslow *diskslow*	**Read, Create, or Append to a File** • Not implemented in Netrexx Pipelines.
filetoken	**Read or Write an SFS File That is Already Open** • Not implemented in Netrexx Pipelines.
fileupdate *diskupdate*	**Change records in a CMS file**
filew *diskw* *>*	**Replace or Create a File** ┌NetRexx┐ ►►--FILEW--*string*--►◄
fillup	**Pass Records To Output Streams** • Not implemented in Netrexx Pipelines.

filterpack	**Manage Filter Packages**	
	• Not implemented in Netrexx Pipelines.	
find	**Select Lines by XEDIT Find Logic**	
| | ```
▶▶-----FIND--+----------+---------▶◀
             +--string--+
``` | |
fitting	**Source or Sink for Copipe Data**	
	• Not implemented in Netrexx Pipelines.	
fmtfst	**Format a File Status Table (FST) Entry**	
	• Not implemented in Netrexx Pipelines.	
frlabel *fromlabel*	**Select Records from the First One with Leading String**	
| | ```
▶▶-----FRLABEL--+---------+--------▶◀
                +--string-+
``` | |
| *fromlabel*
frlabel | **Select Records from the First One with Leading String** | |
| | ```
▶▶-----FROMLABEL--+---------+------▶◀
                  +--string-+
``` | |
| *frtarget* | **Select Records from the First One Selected by Argument Stage** | |
| | ```
▶▶--+--FRTARGET----+--stage--+-----------+------▶◀
    +--FROMTARGet--+         +--operands--+
``` | |
fullscrq	**Write 3270 Device Characteristics**	
	• Not implemented in Netrexx Pipelines.	
fullscrs	**Format 3270 Device Characteristics**	
	• Not implemented in Netrexx Pipelines.	
gate	**Pass Records Until Stopped**	
| | ```
▶▶--GATE--+----------+----------▶◀
          +--STRICT--+
``` | |
gather	**Copy Records From Input Streams**	
	• Not implemented in Netrexx Pipelines.	
getfiles *getfiles* *getfile* *getfil* *getfi* *getf* *get*	**Read Files**	
| | ```
▶▶--GETfiles--------------------▶◀
``` | |
getovers	**Write the Contents of Objects**	
	• Input stream 0 should contain rexx objects. The getovers stage will output the index and contents of the stem on str output stream 1 is connected, the root is placed there. Any severed streams will cause then stage to exit. Passing a object will cause the stage to exit with return code 13. • Pipes for NetRexx only.	
getstems	**Write the Contents of Members of Stems**	
	• Input stream 0 should contain rexx objects containing stems. The getstems stage will output the contents of the ster stream 0. If output stream 1 is connected, the root is placed there. Any severed streams will cause then stage to ex Passing a non rexx stem object will cause the stage to exit with return code 13. • Pipes for NetRexx only.	

grep *regex* *3.09*	**Select Lines by a Regular Expresion** ``` ┌NetRexx───┐ │ │ │ ▶▶──+──GREP───+──+────────────────────────+──regex_Dstring─(1)───▶◀ │ │ +──REGEX──+ +─(──| options_string |──)─+ │ │ │ │ options_string: │ │ ┌────────────────────────────────┐ │ │ |──+─+────────────────────────────────+─+──| │ │ +─Numbers─────────────────────────+ (2) │ │ +─Before─+─1──────+──────────────+ (3) │ │	+─number─+	│ │ +─After─+─1──────+───────────────+ (3) │ │	+─number─+	│ │ +─Context─+─1──────+─────────────+ (4) │ │	+─number─+	│ │ +─NOSeparator────────────────────+ (5) │ │ +─Separator─+─/──/────────────+──+ (5) │ │	+─delimitedString─+	│ │ +─Tertiary───────────────────────+ (6) │ │ +─COUnt──────────────────────────+ (7) │ └───┘ ``` - NetRexx Pipelines only. - Records matching the RegEx are put out on primary output. - Records not matching are put out on secondary, if connected, or discarded. - . - (1) Regex_string is a Java RegEx expresion. Null string passes all records. - (2) Records are prefaced with records number, 10 characters, right justified. - (3) Number of records put out after a matching record. - (4) Number of records put out before and after a matching record. - (5) Inserted before a group of "before records" or the found record with "after records." - (6) Send all matching records (no numbers) to tertiary output stream, if connected. - (7) Only a count of matches is put out on the primary output stream. (Other options probably should not be used wit
hash *dict*	**Read or Write a Dictionary** - Pipes for NetRexx only.
hasha *dicta*	**Read or Write a Dictionary**
hashr *dictr*	**Read or Write a Dictionary**
hashw *dictw*	**Read or Write a Dictionary**
help *?*	**Display Help for Pipelines** - Not implemented in Netrexx Pipelines.
hfs	**Read or Append File in the Hierarchical File System** - Not implemented in Netrexx Pipelines.
hfsdirectory	**Read Contents of a Directory in a Hierarchical File System** - Not implemented in Netrexx Pipelines.

hfsquery	**Write Information Obtained from OpenExtensions into the Pipeline** • Not implemented in Netrexx Pipelines.
hfsreplace	**Replace the Contents of a File in the Hierarchical File System** • Not implemented in Netrexx Pipelines.
hfsstate	**Obtain Information about Files in the Hierarchical File System** • Not implemented in Netrexx Pipelines.
hfsxecute	**Issue OpenExtensions Requests** • Not implemented in Netrexx Pipelines.
hlasm	**Interface to High Level Assembler** • Not implemented in Netrexx Pipelines.
hlasmerr	**Extract Assembler Error Messages from the SYSADATA File** • Not implemented in Netrexx Pipelines.
hole	**Destroy Data** ``` ►►--HOLE---------►◄ ```
hostbyaddr *3.09*	**Resolve IP Address into Domain and Host Name** ``` ►►-------HOSTBYADDR -------------+--------------+------►◄ +--INCLUDEIP----+ (1) ``` • (1) Optional parameter not present in VM/CMS version • INCLUDEIP - Also include the IP address along with the hostname. Output: <hostname>/<ip address> Example: **dns.google/8.8.8.8** • Known issues: The underlying Java method getByName/getHostName does not appear to handle IPv6 addresses i known and consistent manner. Could be related to a host configuration issue but googling shows odd and inconsist results for getting around this.
hostbyname *3.09*	**Resolve a Domain Name into an IP Address** ``` ►►--HOSTBYNAME -------------+--------------+-------►◄ +--INCLUDENAME--+ (1) ``` • (1) Optional parameter not present in CMS Pipelines • Arguments: INCLUDENAME - Also include the name of the host on output. • Output: <hostname>/<ip address> Example: **dns.google/8.8.8.8**
hostid *3.09*	**Write TCP/IP Default IP Address** ``` ►►--HOSTID----+------------------+--------------►◄ +--USERid--word-(1)--+ ``` • (1) The USERid option available under CMS Pipelines is not applicable and is ignored in NetRexx Pipelines
hostname *3.09*	**Write TCP/IP Host Name** ``` ►►--HOSTNAME--+---------------+--+-------------------+---►◄ +--INCLUDEIP--(1)-+ +--USERid--word-(2)--+ ``` • (1) Optional parameter not present in VM/CMS version • (2) The USERid option available under CMS is not applicable and is ignored in NetRexx Pipelines • Arguments: INCLUDEIP - include the IP address of the system in the response in the form <hostname>/<ip address

htmlrows **htmlrow** *3.11*	**Convert rows to HTML format**

```
 NetRexx
                                    +-SEParator ","----+
 ►►--HTMLROWs--+------------+-+------------------+--►
               +-ROW-QString-+ +-SEParator-QString-+

  ►--+------------+-+------------+--►◄
     +-HEAD-QString-+ +-DATA-QString-+
```

- HTMLROWs reads rows from its primary input stream and writes them to its primary output stream, altering them to proper HTML tags for TABLE ROWS.
- I.e., it converts
    abc,mnop,xyz
  into
    <tr><td>abc</td><td>mnop</td><td>xyz</td></tr>
- The SEPARATOR QStirng, by default the comma character, can be specified.
- There are options to put additional data inside the tags. This could be used for class or style tag options, for example
    - ROW QString : puts its information into the <tr>-tags
    - DATA QString : puts its information into the <td>-tags
    - HEAD QString : puts its information into the <th>-tags (1)
- QString is a quoted string of characters. The quote marks may be either single or double, but must match. If there a spaces in the string, the quote marks are optional.
- (1) If there is a HEAD option, the first row read has <th>-tags instead of <td>-tags. It must have a QString of at leas Succeeding rows have the standard <td>-tags.

httpsplit	**Split HTTP Data Stream**

- Not implemented in Netrexx Pipelines.

iebcopy	**Process IEBCOPY Data Format**

- Not implemented in Netrexx Pipelines.

if	**Process Records Conditionally**

- Not implemented in Netrexx Pipelines.

immcmd	**Write the Argument String from Immediate Commands**

- Not implemented in Netrexx Pipelines.

insert	**Insert String in Records**

```
                                +-BEFORE-+
 ►►--INSERT--delimitedString--+-------+---+------------+---►►
                                +-AFTER--+    +-inputRange--+
```

- insert a string into a record before or after the record content. Will be much more efficient than specs especially if th a Byte[]

inside	**Select Records between Labels**

```
 ►►--INSIDE--+------------+--delimitedString--+-number---------+--►◄
             +-ANYcase-----+                  +-delimitedString-+
             +-CASEANY-----+
             +-CASEIGNORE--+
             +-IGNORECASE--+
             +-CASELESS----+
```

instore	**Load the File into a storage Buffer**

- Not implemented in Netrexx Pipelines.

ip2socka	**Build sockaddr_in Structure**

- Not implemented in Netrexx Pipelines.

ispf	**Access ISPF Tables** • Not implemented in Netrexx Pipelines.
jeremy	**Write Pipeline Status to the Pipeline** • Not implemented in Netrexx Pipelines.

join	**Join Records**
	
	<pre> +-1----------------+
	►►--JOIN-+-------+-+------------------+--►
	         +-COUNT-+ +-number-----------+
	                  +-*----------------+
	                  +-KEYLENgth--number-+
	
	 ►--+------------------------------+-+-------+---►◄
	     +-delimitedString-+----------+-+ +-number-+
	                       +-TERMinate-+</pre>

joincont	**Join Continuation Lines**
	
	<pre> +-TRAILING----------+
	►►--JOINCONT--+-----------+-+-----+-+------------------+-+-------+--►
	              +-ANYCase----+ +-NOT-+ +-RANGE--inputRange-+ +-DELAY-+
	              +-CASEANY----+ +-LEADING----------+
	              +-CASEIGNORE-+
	              +-IGNORECASE-+
	              +-CASELESS---+
	
	 ►-+-------+-delimitedString--+------+--+----------------+----►◄
	    +-ANYof-+ +-KEEP-+ +-delimitedString-+</pre>

juxtapose	**Preface Record with Marker**
	
	<pre>►►--JUXTAPOSe----+-------+-------------►◄
	                 +-COUNT-+</pre>

ldrtbls	**Resolve a Name from the CMS Loader Tables** • Not implemented in Netrexx Pipelines.
listcat	**Obtain Data Set Names** • Not implemented in Netrexx Pipelines.
listdsi	**Obtain Information about Data Sets** • Not implemented in Netrexx Pipelines.
listispf	**Read Directory of a Partitioned Data Set into the Pipeline** • Not implemented in Netrexx Pipelines.
listpds	**Read Directory of a Partitioned Data Set into the Pipeline** • Not implemented in Netrexx Pipelines.

listzip	**List the Files in a Zipped File**
	<pre>┌NetRexx─┐
	►►---LISTZIP----zipFileName------►◄</pre>

literal	**Write the Argument String**
	
	<pre>►►---LITERAL---+--------+----------►◄
	               +-string-+</pre>

locate	**Select Lines that Contain a String**

```
▶▶--LOCATE-+-------------+-+----------+--+------------+-+-------+--▶
           +-ANYCase-----+ +-MIXED-(1)-+  +-inputRanges-+ +-ANYof-+
           +-CASEANY-----+ +-ONEs--(1)-+
           +-CASEIGNORE--+ +-ZEROs-(1)-+
           +-IGNORECASE--+
           +-CASELESS----+


 ▶-+-----------------+---▶◀
   +-delimitedString-+
```

(1) Not in NetRexx Pipelines, yet.

lookup	**Find Records in a Reference Using a Key Field**

```
┌─NetRexx────────────────────────────────────────────────────────┐
│                                                                 │
│▶▶--LOOKUP--+-------+--+---------+--+---------+--+--------+-▶      │
│            +-COUNT-+  +-ANYCASE-+  +-AUTOADD-+  +-BEFORE-+       │
│                                                                 │
│ ▶-+---------+--+----------+--+-----------+--+------------+------▶│
│   +-KEYONLY-+  +-SETCOUNT-+  +-INCREMENT-+  +-TRACKCOUNT-+      │
│                                                                 │
│ ▶------------------------+------------------------+---------▶   │
│                          +-inputRange-+-----------+            │
│                                       +-inputRange-+           │
│                                                                 │
│ ▶--+-------------------------+------------------------▶◀        │
│    +-DETAIL MASTER-------------+                                │
│    +-DETAIL ALLMASTER PAIRWISE-+                                │
│    +-DETAIL ALLMASTER----------+                                │
│    +-DETAIL--------------------+                                │
│    +-MASTER DETAIL-------------+                                │
│    +-MASTER--------------------+                                │
│    +-ALLMASTER DETAIL PAIRWISE-+                                │
│    +-ALLMASTER DETAIL----------+                                │
│    +-ALLMASTER-----------------+                                │
└─────────────────────────────────────────────────────────────────┘
```

- in stream 0 are detail records
- in stream 1 are master records
- in stream 2 adds to masters
- in stream 3 delete from masters
-
- out stream 0 are matched records
- out stream 1 are unmatched detail records
- out stream 2 are unmatched or counted master records
- out stream 3 deleted masters
- out stream 4 duplicate masters
- out stream 5 unmatched master deletes
-
- lookup does not consider an unconnected output stream an error. It does proprogate EOFs from output streams.

locate	

lookup	**Find Records in a Reference Using a Key Field**
	<pre>┌─CMS─┐
►►──LOOKUP──┼───────┼─┼─────────────────┼─┼───────────┼──►
           +─COUNT─+ +─MAXcount─*number*─+ +─INCREMENT─+

                                   +─NOPAD────+
  ►──+─────────┼─┼───────────┼─┼─────────┼─┼─────────┼──►
     +─SETCOUNT─+ +─TRACKCOUnt─+ +─PAD─*xorc*─+ +─ANYcase─+

  ►──+───────────────────┼─┼─────────┼─┼────────┼──►
     +─AUTOADD─┼────────┼─+ +─KEYONLY─+ +─STRICT─+
     | +─BEFORE─+ |
     +─CEILING────────────+
     +─FLOOR──────────────+

  ►──+───────────────────────────┼──►
     +─*inputRange*─┼───────────┼─+
                   +─*inputRange*─+

     +─DETAIL MASTER─────────────+
  ►──+───────────────────────────+──────────────────────────►◄
     +─DETAIL ALLMASTER PAIRWISE─+
     +─DETAIL ALLMASTER──────────+
     +─DETAIL────────────────────+
     +─MASTER DETAIL─────────────+
     +─MASTER────────────────────+
     +─ALLMASTER DETAIL PAIRWISE─+
     +─ALLMASTER DETAIL──────────+
     +─ALLMASTER─────────────────+</pre> |
	• in stream 0 are detail records • in stream 1 are master records • in stream 2 adds to masters • in stream 3 delete from masters • • out stream 0 are matched records • out stream 1 are unmatched detail records • out stream 2 are unmatched or counted master records • out stream 3 deleted masters • out stream 4 duplicate masters • out stream 5 unmatched master deletes • • lookup does not consider an unconnected output stream an error. It does proprogate EOFs from output streams.
lookup	**Find Records in a Reference Using a Key Field**
maclib	**Generate a Macro Library from Stacked Members in a COPY File** • Not implemented in Netrexx Pipelines.

mapmdisk	**Map Minidisks Into Data spaces** • Not implemented in Netrexx Pipelines.
mctoasa	**Convert CCW Operation Codes to ASA Carriage Control** • Not implemented in Netrexx Pipelines.
mdiskblk	**Read or Write Minidisk Blocks** • Not implemented in Netrexx Pipelines.
mdskrandom	**Random Access a CMS File on a Mode** • Not implemented in Netrexx Pipelines.
mdskslow	**Read, Append to, or Create a CMS File on a Mode** • Not implemented in Netrexx Pipelines.
mdskupdate	**Replace Records in a File on a Mode** • Not implemented in Netrexx Pipelines.
members	**Extract Members from a Partitioned Data Set** • Not implemented in Netrexx Pipelines.
merge	**Merge Streams** • Not implemented in Netrexx Pipelines.
mqsc	**Issue Commands to a WebSphere MQ Queue Manager** • Not implemented in Netrexx Pipelines.

nfind *notfind*	**Select Lines by XEDIT NFind Logic**

```
►►---+-NFIND---+--+---------+---------►◄
     +-NOTFIND-+  +--string--+
```

ninside *notinside* *3.09*	**Select Records Not between Labels**

```
►►-+-NINSIDE----+-+-----------+-delimitedString-+-number---------+--►◄
   +-NOTINSIDE--+ +-ANYcase----+                 +-delimitedString-+
                 +-CASEANY----+
                 +-IGNORECASE-+
                 +-CASEIGNORE-+
                 +-CASELESS---+
```

nlocate *notlocate*	**Select Lines that Do Not Contain a String**

```
►►-+-NLOCATE---+-+------------+--+----------+--+------------+-----►
   +-NOTLOCATE-+ +-ANYCase-----+  +-MIXED-(1)-+  +-inputRanges-+
                +-CASEANY-----+  +-ONEs--(1)-+
                +-CASEIGNORE--+  +-ZEROs-(1)-+
                +-IGNORECASE--+
                +-CASELESS----+

►-+-------+-+---------------+-►◄
  +-ANYof-+ +-delimitedString-+
```

• (1) Not in NetRexx Pipelines, yet.

noEofBack	**Pass Records and Ignore End-of-file on Output**

```
►►--NOEOFBACK---------►◄
```

nop	**No Operation**
	
	``
	  ┌─NetRexx─
	  │
	▶▶──NOP─────────▶◀
	``
	
	• Pipes for NetRexx only.

not	**Run Stage with Output Streams Inverted** ``` ▶▶──NOT──stage──+───────────+──────▶◀ +──operands──+ ```
notfind *nfind*	**Select Lines by XEDIT NFind Logic** ``` ▶▶───+─NOTFIND─+──+──────────+────────▶◀ +─NFIND───+ +──string──+ ```

notinside
ninside
	Select Records Not between Labels
	
	``
	▶▶─+─NOTINSIDE──+─+───────────+─delimitedString─+─number─────────+──▶◀
	   +─NINSIDE────+ +─ANYcase────+ +─delimitedString─+
	                   +─CASEANY────+
	                   +─IGNORECASE─+
	                   +─CASEIGNORE─+
	                   +─CASELESS───+
	``

notlocate
nlocate
	Select Lines that Do Not Contain a String
	
	``
	▶▶─+─NOTLOCATE─+─+────────────+──+──────────+──+─────────────+────▶
	   +─NLOCATE───+ +─ANYCase─────+ +─MIXED─(1)─+ +─inputRanges─+
	                 +─CASEANY─────+ +─ONEs──(1)─+
	                 +─CASEIGNORE──+ +─ZEROs─(1)─+
	                 +─IGNORECASE──+
	                 +─CASELESS────+
	
	▶─+───────+─+────────────────+─▶◀
	  +─ANYof─+ +─delimitedString─+
	``
	
	• (1) Not in NetRexx Pipelines, yet.

nucext	**Call a Nucleus Extension** • Not implemented in Netrexx Pipelines.
optcdj	**Generate Table Reference Character (TRC)** • Not implemented in Netrexx Pipelines.

outside	**Select Records Not between Labels**
	
	``
	▶▶──OUTSIDE─+────────────+─delimitedString─+─number─────────+──▶◀
	           +─ANYcase────+ +─delimitedString─+
	           +─CASEANY────+
	           +─CASEIGNORE─+
	           +─IGNORECASE─+
	           +─CASELESS───+
	``

outstore	**Unload a File from a storage Buffer** • Not implemented in Netrexx Pipelines.
over	**Write the Values of Stems** • Obsolete. Now use varover. over is now an alias for overlay..

overlay *overla* *overl* *over*	**Overlay Data from Input Streams** ``` ┌─NetRexx───┐ │ +-NOHOLD-(1)-+ +-PAD-(1)+ +-BLANK----+ │ │ ▶▶--OVERlay-----+-----------+-+--------+-+----------+-----▶ │ │ +-HOLD-(1)---+ +-xorc----+ │ │ +-SPACE-(1)+ │ │ │ │ │ │ ▶--+------------------------+-+----------------------------+--▶◀ │ │ +-TRANSparent-+-xorc--+-(1)+ +-STRING--delimitedString-(1)(2)+ │ │ +-BLANK-+ │ │ +-SPACE-+ │ └───┘ ``` - HOLD keeps the last record from each stream, except primary, and uses it if the stream ends. - TRANSPARENT means that character can be different from the PAD character. If omitted, it is the same as PAD character. - dstream can be used instead of a non-primary stream. - (1) NetRexx Pipelines only - (2) same as highest (+1) stream; implies HOLD
overlay	**Overlay Data from Input Streams** ``` ┌─CMS─────────────────────────────────┐ │ +-BLANK-+ │ │ ▶▶--OVERlay-+-------+---------▶◀ │ │ +-xorc--+ │ └─────────────────────────────────────┘ ```
overstr	**Process Overstruck Lines** - Not implemented in Netrexx Pipelines.
pack	**Pack Records as Done by XEDIT and COPYFILE** - Not implemented in Netrexx Pipelines.

pad	**Expand Short Records**
	
	``
	      +-Right-+ +-BLANK-+
	 ▶▶-PAD-+-------+--+-----------------+--number--+-------+--▶◀
	      +-Left--+ +-+--------+-MODULO-+ +-xorc--+
	                    +-number-+
	``

parcel	**Parcel Input Stream Into Records** - Not implemented in Netrexx Pipelines.

parse
3.09
	Rearrange Contents of Records
	
	``
	┌─NetRexx───┐
	│ ▶▶--PARSE--parse_template_Dstring---output_template_Dstring--▶◀ │
	└───┘
	``
	
	- Records are parsed via the parse_template_delimited_string.
	- Variables are named $n, where n is 1 to 9.
	- The values of the variables are put into the output_template_delimited_string replacing $n.
	- For a literal $n that won't be changed, use $$n.
	- NetRexx Pipelines only.
	- .
	- Example:
	  `parse / 2 $1 +1/ /The second letter is "$1". $$1 won't be changed./`

pause	**Signal a Pause Event** - Not implemented in Netrexx Pipelines.
pdsdirect	**Write Directory Information from a CMS Simulated Partitioned Data Set** - Not implemented in Netrexx Pipelines.

pick	**Select Lines that Satisfy a Relation**

```
┌─NetRexx─────────────────────────────────────────────────┐
│                                                          │
│            +-NOPAD----+                                  │
│ ▶▶--PICK-+----------+-+------------+--▶                   │
│            +-PAD-xorc-+ +-ANYcase-(1)+                    │
│                        +-CASEANY----+                    │
│                        +-CASEIGNORE-+                    │
│                        +-CASELESS---+                    │
│                        +-IGNORECASE-+                    │
│                                                          │
│                                                          │
│  ▶--+------------+-+-==--+-+-----------------+-▶◀        │
│      +-inputRange-+ +-^==-+ +-inputRange------+           │
│                    +-<<--+ +-delimitedString-+           │
│                    +-<<=-+                                │
│                    +->>--+                                │
│                    +->>=-+                                │
│                    +-\==-+(2)                             │
│                    +-/==-+                                │
│                    +-=---+                                │
│                    +-^=--+                                │
│                    +-<---+                                │
│                    +-<=--+                                │
│                    +->---+                                │
│                    +->=--+                                │
│                    +-\=--+(2)                             │
│                    +-/=--+                                │
│                                                          │
└──────────────────────────────────────────────────────────┘
```

- (1) Can be before PAD/NOPAD. Depreciated.
- (2) The backslash (\) may need to be escaped, doubled, in some systems shells.

pick	**Select Lines that Satisfy a Relation**

```
 CMS
               +-NOPAD-----+
►►--PICK---+-----------+--+------------+----►
               +-PAD--xorc-+  +-ANYcase----+
                               +-CASEANY----+
                               +-CASEIGNORE-+
                               +-CASELESS---+
                               +-IGNORECASE-+


 ►--+--+--------------------+---| List |-+--►◄
     |   +-+-FROM-+--+-------+-+           |
     |   | +-TO---+  +-AFTER-+ |           |
     |   +-WHILE--------------+           |
     +-| Fromto |-------------------------+


 Fromto:
 |--FROM--+-------+--| List |--+-TO-+-------+--| List |-+--|
          +-AFTER-+            |    +-AFTER-+            |
                              +-COUNT--number----------+


 List:
 |--+-------------------+--| Test |--|
    +-| List |--+-AND-+-+
                +-OR--+


 Test:
 |--| RangeString |--+--| NonEqualOp |--| RangeString |--+--|
                     +--| EqualOp |----| CommaList |-----+


 CommaList:
     ⌐---,---------------+
 |-----| RangeString |--+--|


 RangeString:
 |--+-inputRange-------+--|
    +-delimitedString--+
    +-number+----------+


 Character Operators::
  == ^== \== /== << <<= >> >>= IN NOTIN


 Numeric Operators:
  = ^= < <= > >=
```

| *pickparse*<br>*3.09* | **Select Lines that Satisfy Relations using Rexx Parse**<br><br>┌─**NetRexx**─────────────────────────────────────────────────────────────────────┐<br>```
                     +--ONE---------------+                    ┌-- (2)-----------------+
 ►►--PICKPARSE--+--------------------+--parse_Dstring-----+-----------------+--+--►◄
                     +--ALL--+----------+--+                    +--logic_Dstring--+
                             +--SINGLE--+                       +--ELSE-(1)-------+
```<br><br>• Records are parsed via the parse_delimited_string.<br>• Variables are named $n, where n is 1 to 9.<br>• The values of the variables are put into the logic_delimited_string replacing $n and evaluated. If TRUE, the record is<br>  on the stream numbered by the dstring's position.<br>• The stream for a Dstring of ELSE is used if no previous logic Dstring is TRUE.<br>• If there is no specific ELSE, there is an implied one at the end; if that stream is not connected, the record is discarded<br>• If ONE then the record is put out on, at most, one stream: the first one matched.<br>• If ALL then the record is put out on all streams matched.<br>• If SINGLE then the records are all put out on the primary output stream.<br>• The parse_delimited_string and logic_delimited_string(s) follow normal NetRexx rules.<br>• (1) Implied ELSE after last specified dstring.<br>• (2) Up to 10 logic_Dstrings may be specified to go to up to 11 ouput streams (including an implied ELSE).<br>• Not implemented in CMS Pipelines.<br><br>Pickparse permits selecting records by a NetRexx logical expression, using parts of the record selected by a Rexx PAR<br>template.<br><br>A simple example has two delimited strings, a Rexx template and a logical expression:<br><br>`pickparse / . . $3 . 50 $5 +5 / / $3 < $5 /`<br><br>The parse template selects the 3rd word, and the 5 characters starting in column 50. the variable names are a dollar sig<br>digit. Then those variables can be used in the logic expression. When run, and records matching the logic expression a<br>to the primary output stream, others to the secondary. If either stream is not connected, the corresponding records are<br>discarded.<br><br>There can be multiple logic expressions, each in its own delimited string. Parenthetical expressions may be used. Reco<br>matched to each in turn. Any records matching are written to that output stream, if connected.<br><br>With the option ONE, the default, each record is written to one output stream: the first one it matches. With the option A<br>matching goes on and a record could be written to multiple output streams.<br><br>There is an implicit or explicit ELSE as the last logic expression. Records that have not matched any of the previous ex<br>match this and are written or discarded depending on if the stream is connected or not.<br><br>The parse template can define up to 9 separate zones, $1 to $9. The variables $_n are also available for the logic expre<br>they are the values from the previous record. Initially these are "".<br><br>There can be up to 10 output streams defined, and up to 9 logic expressions plus ELSE. |
pipcmd	**Issue Pipeline Commands** • Not implemented in Netrexx Pipelines.
pipestop	**Terminate Stages Waiting for an External Event** • Not implemented in Netrexx Pipelines.
polish	**Reverse Polish Expression Parser** • Not implemented in Netrexx Pipelines.
predselect	**Control Destructive Test of Records** • Not implemented in Netrexx Pipelines.
preface	**Put Output from a Device Driver before Data on the Primary Input Stream** • Not implemented in Netrexx Pipelines.

prefix	**Stop and Run a Stage First, Before Continuing** ``` ┌NetRexx─┐ ▶▶--PREFIX---string-------------▶◄ ``` • Blocks its primary input and excutes stage supplied as an argument. The output from this stage are put to the prima stream. When its compete the primary input is shorted. • Not implemented in CMS Pipelines.
printmc	**Print Lines** • Not implemented in Netrexx Pipelines.
punch	**Punch Cards** • Not implemented in Netrexx Pipelines.
qpdecode	**Decode to Quoted-printable Format** • Not implemented in Netrexx Pipelines.
qpencode	**Encode to Quoted-printable Format** • Not implemented in Netrexx Pipelines.
qsam	**Read or Write Physical Sequential Data Set through a DCB** • Not implemented in Netrexx Pipelines.

query	**Obtain Information From Pipelines**
	
	``
	              +-VERSION------+
	▶▶--Query--+-------------+----------▶◄
	              +-LEVEL--------+
	              +-SOURCE-(1)---+
	              +-MSGLEVEL-(2)-+
	              +-MSGLIST-(2)--+
	``
	
	• (1) Not CMS
	• (2) Not NetRexx Pipelines

random
3.09
	Generate Pseudorandom Numbers
	
	``
	▶▶--RANDOM--+----------------------------------+----▶◄
	
		+-*----------+ +-*-----------+ |
		            +-+-max_number-+--+-------------+-+
		                              +-seed_snumber-+
		``
		
		• NetRexx Pipelines will be a different sequence than CMS gives with the same seed.

reader	**Read from a Virtual Card Reader** • Not implemented in Netrexx Pipelines.
readpds	**Read Members from a Partitioned Data Set** • Not implemented in Netrexx Pipelines.

regex *grep* *3.09*	**Select Lines by a Regular Expresion**

```
┌─NetRexx─────────────────────────────────────────────────────┐
│                                                             │
│ ►►──+──REGEX──+──+─────────────────────────+──regex_Dstring─(1)───►◄ │
│      +──GREP───+  +─(──| options_string |──)─+                │
│                                                             │
│ options_string:                                             │
│     ┌────────────────────────────────┐                      │
│ |──+─+────────────────────────────────+─+──|                 │
│       +─Numbers───────────────────────+ (2)                  │
│       +─Before─+─1──────+─────────────+ (3)                  │
│       |        +─number─+             |                      │
│       +─After─+─1──────+──────────────+ (3)                  │
│       |        +─number─+             |                      │
│       +─Context─+─1──────+────────────+ (4)                  │
│       |          +─number─+           |                      │
│       +─NOSeparator───────────────────+ (5)                  │
│       +─Separator─+─/──/────────────+──+ (5)                 │
│       |            +─delimitedString─+ |                     │
│       +─Tertiary──────────────────────+ (6)                  │
│       +─COUnt─────────────────────────+ (7)                  │
└─────────────────────────────────────────────────────────────┘
```

- NetRexx Pipelines only.
- Records matching the RegEx are put out on primary output.
- Records not matching are put out on secondary, if connected, or discarded.
- .
- (1) Regex_string is a Java RegEx expresion. Null string passes all records.
- (2) Records are prefaced with records number, 10 characters, right justified.
- (3) Number of records put out after a matching record.
- (4) Number of records put out before and after a matching record.
- (5) Inserted before a group of "before records" or the found record with "after records."
- (6) Send all matching records (no numbers) to tertiary output stream, if connected.
- (7) Only a count of matches is put out on the primary output stream. (Other options probably should not be used wit

retab	**Replace Runs of Blanks with Tabulate Characters** • Not implemented in Netrexx Pipelines.

reverse	**Reverse Contents of Records**

```
┌───────────────────────────────────────┐
│ ►►──REVERSE────────────────────►◄       │
└───────────────────────────────────────┘
```

rexx	**Run a REXX Program to Process Data** • Not implemented in Netrexx Pipelines.
rexxvars	**Retrieve Variables from a REXX or CLIST Variable Pool** • Not implemented in Netrexx Pipelines.
runpipe	**Issue Pipelines, Intercepting Messages** • Not implemented in Netrexx Pipelines.
scm	**Align REXX Comments** • Not implemented in Netrexx Pipelines.
sec2greg	**Convert Seconds Since Epoch to Gregorian Timestamp** • Not implemented in Netrexx Pipelines.

serialize	**Convert Objects to a Single Text String**

```
┌─NetRexx───────────────────────────┐
│                                   │
│ limited information available     │
└───────────────────────────────────┘
```

- {class} if class is specified deserialize input to objects of this type otherwise serialize input objects.
- Pipes for NetRexx only.
- For some reason readObject does not like more than one object network in its stream. Block multiple objects. see examples/sertest.njp

sfsback	**Read an SFS File Backwards**	
	• Not implemented in Netrexx Pipelines.	
sfsdirectory	**List Files in an SFS Directory**	
	• Not implemented in Netrexx Pipelines.	
sfsrandom	**Random Access an SFS File**	
	• Not implemented in Netrexx Pipelines.	
sfsupdate	**Replace Records in an SFS File**	
	• Not implemented in Netrexx Pipelines.	

---

snake
3.09

Build Multicolumn Page Layout

```
►►--SNAKE--number_cols--+------------------------------------------------+--►◄
                        +--number_rows--+----------------------------+--+
                                        +--page_seperator_DString-(1)-+
```

- (1) NetRexx Pipelines only. Appears first, last, and between pages.
  Avoid \ as escape terms maybe added in the future. \n for newline is OK.
  Your system may require \\n .

---

socka2ip

Format sockaddr_in Structure

- Not implemented in Netrexx Pipelines.

---

sort

Order Records

```
 NetRexx
►►--SORT--+----------------------------+--+------------+--►
          |   +-REXX-----+  +-10000-+   |  +-inputRange-+
          +-(-+----------+--+-------+-)-+
              +-class-(2)+  +-size--+

    +-Ascending-(1)-+
 ►--+--------------+--+------------+--►◄
    +-Descending-(1)+  +-SINGLEOK-(3)+
```

- (1) May come before inputRange, for backwards compatability.
- (2) Requires that you implement another sortClass with a name begining with 'sort'
- (3) Suppresses error message if only one record to sort for Rexx objects.
- Uses *sortClass* class as Interface Class for Generic Sort Objects
  and *sortRexx* class to Sort Rexx Text Objects

---

sort

Order Records

```
 CMS
                         +-NOPAD----+
►►--SORT-+--------+--+---------+--+---------+---►
         +-COUNT--+  +-PAD-xorc-+  +-ANYcase-+
         +-UNIQue-+

    +-Ascending-----------------------------------+
 ►--+---------------------------------------------+---►◄
    +-Descending----------------------------------+
    | +-----------------------------------------+  |
    | |                 +-Ascending--+          |  |
    +-v-inputRange-+-----------+--+---------+-+--+
                   +-Descending-+  +-NOPAD----+
                                   +-PAD-xorc-+
```

space
3.09
	Space Words Like REXX
	
	``
	                                      (1)
	          +-1------+ +-BLANK----------------------+
	►►--SPACE-+--------+-+----------------------------+--►
	          +-number-+ +-xorc-----------------------+
	                     +-+-------+-delimitedString-+
	                       +-STRing-+
	
	          (2)
	        +-BLANK------------------------+
	►---+----------------------------------+-►◄
	    +-xorc-------------------------+
	
		+-ANYof-----+	
			    +-+-----------+-delimitedString-+
			      +-ALLof-(3)-+
			``
			
			- (0) The order is the reverse of CHANGE!
			- (1) the replacement char/string
			- (2) the char/chars that will be stripped and replaced
			- (3) NetRexx Pipelines only, not CMS. The dstring is treated as a single unit for stripping or replacing

spec
specs
	Rearrange Contents of Records
	
	``
	                     +-STOP--ALLEOF----+ (3)
	►►--SPECs--+-----------------+--------►
	                     +-STOP-+-ANYEOF-+-+ (3)
	                     +-n------+ (3)
	
	      ⌐---+
	 ►--+--+-| Group |--------------------------+-+--►◄
	         +-READ-------------------------------+ (5)
	         +-READSTOP---------------------------+
	         +-WRITE------------------------------+
	         +-SELECT-+-streamnum-+---------------+
	
		+-streamid--+	(3)
			
				+-FIRST-----+ |
				
					+-SECOND----+ |
					         +-PAD--+-char----+-------------------+
					
						+-hexchar-+ |
						
							+-BLANK---+ |
							
								+-SPACE---+ |
								         +-+-WORDSEParator---+---+--char---+--+ (3)
								           +-WS------------ + +-hexchar-+ (3)
								           +-FIELDSEparator- + +-BLANK---+ (3)
								           +-FS------------ + +-SPACE---+ (3)
								
								Group:
								 |--| Input |--| Conversion |--| Output |--| Alignment |--|
								
								Input:
								 |--+-Words-(1)-wnumberrange----------------------+-------|
								    +-Fields-(1)-fnumberrange--------------------+ (3)
								    +-cnumberrange-------------------------------+
								    +-delimitedString----------------------------+
								    +-Xhexstring---------------------------------+
								    +-Hhexstring---------------------------------+
								    +-Bbinstring---------------------------------+
								
									+-FROM--1-------+ +-BY--1-----+ |
									    +-+-RECNO--+-+-----------------+-+-----------+--+
									
										+-NUMBER-+ +-FROM--fromnum-+ +-BY--bynum-+ |
										    +-TODclock-----------------------------------+
										``

```
Conversion:
  |-+-------+--+------------------+------|
    +-STRIP-+  +-B2C----------------+
                +-B2D----------------+ (4)
                +-B2X----------------+ (4)
                +-C2B----------------+
                +-C2D----------------+
                +-C2F----------------+ (3)
                +-C2I----------------+ (3)
                +-C2P-+------------+-+ (3)
                |     +-(2) (scale)-+ | (3)
                +-C2V----------------+ (3)
                +-C2X----------------+
                +-D2C----------------+
                +-D2X----------------+ (4)
                +-F2C----------------+ (3)
                +-I2C----------------+ (3)
                +-P2C-+------------+-+ (3)
                |     +-(2) (scale)-+ | (3)
                +-V2C----------------+ (3)
                +-X2B----------------+ (4)
                +-X2C----------------+
                +-X2D----------------+ (4)
                +-f2t----------------+
                +-LOWER--------------+ (4)
                +-UPPER--------------+ (4)
                +-STRING-------------+ (4)
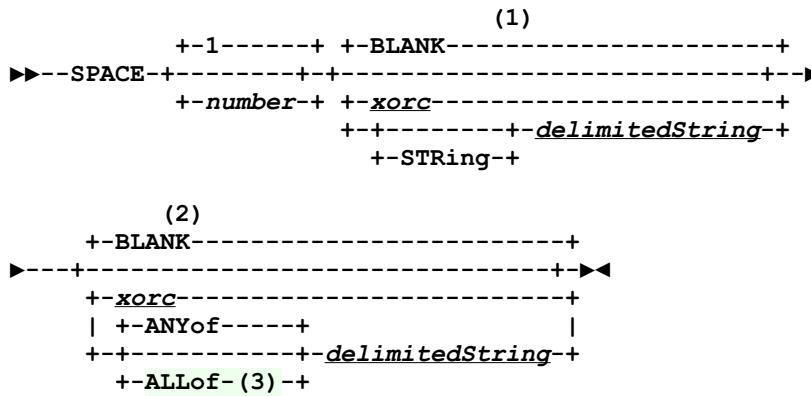

Output:
  |--+-Next-+-------+------+--|
     |       +-(2).n-+        |
     +-NEXTWord-+-+-------+-+
     +-NWord---- + +-(2).n-+ |
     +-columnrange---------+


Alignment:
  |--+--------+--|
     +-Left---+
     +-Center-+
     +-Centre-+
     +-Right--+


Ranges (cnumberrange,  fnumberrange (3),  wnumberrange):
  |--+-snumber-+--(2)--+----------------------+--|
     +---*-----+        +--.-----(2)----number----+
                        +- - -+--(2)--+-snumber-+-+
                        +--;--+        +---*-----+
```

- (1) Blanks are optional in this position.
- (2) Blanks are not allowed here.
- (3) CMS only. Not yet implemented in NetRexx Pipelines
- (4) NetRexx Pipelines only. Not yet implemented in CMS
- (5) READ is giving the same output as READSTOP when the streams are different length.
- [6] This senses if it is the first stage, but comment stages will fool it into not producing any output.

spill	**Spill Long Lines at Word Boundaries**
	• Not implemented in Netrexx Pipelines.

split	**Split Records Relative to a Target**
	
	``
	►►--SPLIT--+-------------+--+------------------+------►
	          +-ANYcase-----+ +--MINimum--number--+
	          +-CASEANY-----+
	          +-CASEIGNORE--+
	          +-IGNORECASE--+
	          +-CASELESS----+
	
	     +-AT-------------------+ +-BLANK---------------+
	 ►--+----------------------+--+-----+--+---------------------+--►◄
	     +--+---------+--+-BEFORE-+ +-NOT-+ +-| target |-+--------+-+
	        +-snumber-+ +-AFTER--+ +-number-+
	
	target:
	    |--+--xrange-------------------------+--|
	       +--+--STRing--+---delimitedString--+
	          +--ANYof---+
	``

| *sql* <br> *3.09* | **Interface to SQL** <br><br> ```
                                                    +-;-+
►►--SQL--+----------------+--+--------------------------+-+---+--►◄
         +-(| options |)-+  +-sql_statement_string-(3)+

options:
   ┌--------------------------------------------+
 |---+--------------------------------------------+-+-|
     |                +-/sqlselect.properties/-+     |
     +-PROPERTIES-+-filename_Qword-(7)-----+-(5)+-+
     | +-HEADERS---+                              |
     +-+               +-(5)(6)--------------------+
     | +-NOHEADERS-+                              |
     +-COUNT2SECondary-(5)(11)------------------+
     +-URL-Qword-(5)(7)--------------------------+
     +-JDBCDRIVER-Qword-(5)(7)------------------+
     +-DBMS-Qword-(5)(7)(8)--------------------+
     +-DB_NAME-Qword-(5)(7)(8)------------------+
     +-USER-Qword-(5)(7)(8)(10)----------------+
     +-PASS-Qword-(5)(7)(8)(10)----------------+
``` <br><br> • uses jdbc to select from any jdbc enabled dbms <br> • properties file (sqlselect.properties default) is read from the secondary input stream to find jdbcdriver name, url, use <br> • sample properties file: <br> • <br> ```
#JDBC driver name
#Tue Feb 03 23:29:43 GMT+01:00 1998
jdbcdriver=com.imaginary.sql.msql.MsqlDriver
url=jdbc:msql://localhost:1114/TESTDB
# the following are not needed for some DBMS, ex: SQLite
user=db_user_name
pass=password_for_db
``` <br> • <br> • if this file is not found default (compiled in) values are used <br> • (1) when using a sql select * (all columns) from the commandline, quote the query as in <br>     **java pipes.compiler (query) "sql select * from dept | console"** <br> • (2) the netrexx/jdbc combination is extremely case sensitive for column and table names <br> • (3) this sql_select_string executed, then statements are read from the primary input stream. <br>    this is optional in NetRexx Pipelines only. <br> • (4) CMS does not use the stream input <br> • (5) NetRexx Pipelines only <br> • (6) CMS Pipelines is implyed HEADERS only. <br> • (7) A Qword is an optionally quoted word. If it contains spaces, it must be quoted. <br> • (8) EXPERIMENTAL Subject to change. DBMS is the kind of database, e.g. SQLite. DB_name is the file name. The <br>    used in place of URL and JDBCDRIVER. SQLite is the only one tested as of 8/15/20. <br> • (9) the SQLSELECT stage uses HEADERS as the default. <br> • (10) USER & PASS are needed for some DBMSs and not others, ex. SQLite. <br> • (11) the count or other output from non-select statements goes to the secondary output stream if connected, or is di <br>    Otherwise it goes to the primary. <br> • <br> • Priority order for URL, JDBCDRIVER and DBMS, DB_NAME (first one found rules): <br>    1. option in the SQL command string <br>    2. from secondary input stream <br>    3. from "sql.properties" file or from file specified by PROPERTIES option <br>    4. Builtin |
| *sqlcodes* | **Write the last 11 SQL Codes Received**

 • Not implemented in Netrexx Pipelines. |

sqlselect	**Query a Database and Format Result**

```
▶▶--SQLSELECT--+---------------+--▶
              +-(-| options |-)-+

   +-SELECT-+                                        +-;-+
 ▶-+--------+-+-------------------------------+-+---+--▶◀
              +-sql_select_statement_string-(3)-+

 options:
    ┌--------------------------------------------+
  |---+------------------------------------------+-+-|
      |                 +-/sqlselect.properties/-+   |
      +-PROPERTIES-+-filename_Qword-(7)-----+-(5)+-+
      | +-NOHEADERS-+                               |
      +-+           +-(5)(6)--------------------+
      | +-HEADERS---+                               |
      +-URL-Qword-(5)(7)-------------------------+
      +-JDBCDRIVER-Qword-(5)(7)------------------+
      +-DBMS-Qword-(5)(7)(8)---------------------+
      +-DB_NAME-Qword-(5)(7)(8)------------------+
      +-USER-Qword-(5)(7)(8)(10)-----------------+
      +-PASS-Qword-(5)(7)(8)(10)-----------------+
```

- (1) when using a sqlselect * (all columns) from the commandline, quote the query as in java pipes.compiler (query) "
  * from dept | console"
- (2) the netrexx/jdbc combination is extremely case sensitive for column and table names
- (3) if no sql_select_string is specified, it is read from the primary input stream.
  this is optional in NetRexx Pipelines only. CMS does not use the stream input.
- (4) a maximum of only one record is ever read from the primary input stream.
- (5) NetRexx Pipelines only
- (6) CMS Pipelines is implied HEADERS only.
- (7) A Qword is an optionally quoted word. If it contains spaces, it must be quoted.
- (8) EXPERIMENTAL Subject to change. DBMS is the kind of database, e.g. SQLite. DB_name is the file name. The
  used in place of URL and JDBCDRIVER. SQLite is the only one tested as of 8/15/20.
- (9) the SQL stage uses NOHEADERS as the default.
- (10) USER & PASS are needed for some DBMSs and not others, ex. SQLite.
- Priority order for URL, JDBCDRIVER, DBMS, DB_NAME, USER, & PASS (first one found rules):
  1. option in the SQL command string
  2. from secondary input stream
  3. from "sqlselect.properties" file or from file specified by PROPERTIES option
  4. Builtin

stack	**Read or Write the Program Stack**

- Not implemented in Netrexx Pipelines.

starmon	**Write Records from the *MONITOR System Service**

- Not implemented in Netrexx Pipelines.

starmsg	**Write Lines from a CP System Service**

- Not implemented in Netrexx Pipelines.

starsys	**Write Lines from a Two-way CP System Service**

- Not implemented in Netrexx Pipelines.

state	**Provide Information about CMS Files** • Not implemented in Netrexx Pipelines.
state	**Verify that Data Set Exists** • Not implemented in Netrexx Pipelines.
statew	**Provide Information about Writable CMS Files** • Not implemented in Netrexx Pipelines.
stem	**Retrieve or Set Variables in a REXX or CLIST Variable Pool** ``` ┌NetRexx┐ ►►--STEM--stem-------------►◄ ```

stem	**Retrieve or Set Variables in a REXX or CLIST Variable Pool**
	
	``
	┌CMS┐
	►►--STEM--stem--+---------+-+-------+-+---------+-►
	                +-PRODUCER-+ +-number-+ +-NOMSG233-+
	                +-MAIN-----+
	      +-SYMBOLIC-+
	   ►-+---------+-+------------+----►◄
	      +-DIRECT---+ +-APPEND-------+
	                  +-FROM--number-+
	``

stfle	**Store Facilities List** • Not implemented in Netrexx Pipelines.
storage	**Read or Write Virtual Machine Storage** • Not implemented in Netrexx Pipelines.
strasmfind	**Select Statements from an Assembler File as XEDIT Find** • Not implemented in Netrexx Pipelines.
strasmnfind	**Select Statements from an Assembler File as XEDIT NFind** • Not implemented in Netrexx Pipelines.

strfind	**Select Lines by XEDIT Find Logic**
	
	``
	►►----STRFIND--+-----------+--delimitedString---------►◄
	              +-ANYcase----+
	              +-CASEANY----+
	              +-IGNORECASE-+
	              +-CASEIGNORE-+
	              +-CASELESS---+
	``
	
	• Not implemented in Netrexx Pipelines.

strfrlabel
strfrlabe
strfrlab
strfrlab
	Select Records from the First One with Leading String
	
	``
	      +--STRFROMLABEL--+ +-INCLUSIVe-+
	►►--+--STRFRLABel----+-+-----------+-+----------+-delimitedString--►◄
	                       +-ANYcase----+ +-EXCLUSIVe-+
	                       +-CASEANY----+
	                       +-IGNORECASE-+
	                       +-CASEIGNORE-+
	                       +-CASELESS---+
	``

|	

strfromlabel
strfrlabel
strfrlabe
strfrlab
	Select Records from the First One with Leading String
	
	``
	        +--STRFROMLABEL--+ +-INCLUSIVe-+
	►►--+--STRFRLABel----+-+-----------+-+----------+-delimitedString--►◄
	                        +-ANYcase----+ +-EXCLUSIVe-+
	                        +-CASEANY----+
	                        +-IGNORECASE-+
	                        +-CASEIGNORE-+
	                        +-CASELESS---+
	``

strip	**Remove Leading or Trailing Characters**
	
	``
	                            +-BOTH-----+
	  ►►--STRIP--+-----------+--+----------+--+-----+--►
	             +--| case |--+ +-LEADING--+ +-TO--+
	                            +-TRAILING-+ +-NOT-+
	
	      +-BLANK---------------------+
	  ►--+---------------------------+--►◄
	      +--| target |--+----------+--+
	                     +--number--+ (1)
	
	  case:
	  |--+-------------+--|
	     +--ANYCase-----+
	     +--CASEANY-----+
	     +--CASEIGNORE--+
	     +--IGNORECASE--+
	     +--CASELESS----+
	
	  target:
	  |--+--xrange-----------------------+--|
	     +--+--STRing--+--delimitedString--+
	        +--ANYof---+
	``
	
	• (1) Not implemented in Netrexx Pipelines.

strliteral	**Write the Argument String**
	
	``
	►►--STRLITeral--+----------------------------+-+----------------+--►◄
	
		+-PREFACE-+	+-delimitedString-+
			                +-+---------+--+-----------+-+
			
				+-APPEND--+ +-CONDitional-+ |
				                +-IFEMPTY--------------------+
				``

strnfind	**Select Lines by XEDIT NFind Logic**
	
	``
	►►--STRNFIND--+-----------+--delimitedString---------►◄
	              +-ANYcase----+
	              +-CASEANY----+
	              +-IGNORECASE-+
	              +-CASEIGNORE-+
	              +-CASELESS---+
	``

strtolabel
strtolabe
strtolab
	Select Records to the First One with Leading String
	
	``
	                                +-INCLUSIVe-+
	►►--STRTOLABel--+-----------+--+----------+--delimitedString------►◄
	                +-ANYcase----+ +-EXCLUSIVe-+
	                +-CASEANY----+
	                +-IGNORECASE-+
	                +-CASEIGNORE-+
	                +-CASELESS---+
	``

structure	**Manage Structure Defnitions** • Not implemented in Netrexx Pipelines.

strwhilelable
strwhilelabl
strwhilelab
strwhilela
strwhilel
strwhile
3.09
	Select Run of Records with Leading String
	
	``
	                                  +-INCLUSIVe-+
	►►--STRWHILElabel--+-----------+--+----------+-delimitedString--►◄
	                   +-ANYcase----+ +-EXCLUSIVe-+
	                   +-CASEANY----+
	                   +-IGNORECASE-+
	                   +-CASEIGNORE-+
	                   +-CASELESS---+
	``

stsi	**Store System Information** • Not implemented in Netrexx Pipelines.
subcom	**Issue Commands to a Subcommand Environment** • Not implemented in Netrexx Pipelines.
substring	**Write substring of record** • Not implemented in Netrexx Pipelines.
synchronise *synchronize*	**Synchronise Records on Multiple Streams** • Not implemented in Netrexx Pipelines.
synchronize *synchronise*	**Synchronise Records on Multiple Streams** • Not implemented in Netrexx Pipelines.
sysdsn	**Test whether Data Set Exists** • Not implemented in Netrexx Pipelines.
sysout	**Write System Output Data Set** • Not implemented in Netrexx Pipelines.
sysvar	**Write System Variables to the Pipeline** • Not implemented in Netrexx Pipelines.
tag *3.11*	**Surrounds Input Records with a HTML tag and its End Tag** ``` ┌─NetRexx───────────────────────── ►►--TAG----string-------------------►◄ ``` • Outputs a record: <tag string>, then passes through all records on its primary input, and finally a record: </tag>. • String may be empty.

tags
3.11
	Surrounds Input Records with HTML tags and their End Tags
	
	┌─**NetRexx**───┐
	▶▶--TAGS----delimitedString--+---------------------+---▶◀
	                                           │ ┌─────────────────┐ │
	                                           +--+-*delimitedString*-+-+
	
	- Outputs a record for each delimitedString: \<delimitedString>, then passes through all records on its primary input, a a record for each, in reverse order: \</first_word_of_delimitedString>.
	- Any delimitedString may be a single word.

take	**Select Records from the Beginning or End of the File**
	
	``
	              +-FIRST-+ +-1----------+
	▶▶--TAKE--+-------+--+------------+--+----------+-------------▶◀
	              +-LAST--+ +-number-----+ +-BYTEs(1)-+
	                         +-snumber(2)-+
	                         +-*----------+
	``
	
	- (1) CMS must be BYTES
	- (2) Not CMS; NetRexx Pipelines: minus reverses first/last

tape	**Read or Write Tapes** - Not implemented in Netrexx Pipelines.
tcpchsum	**Compute One's complement Checksum of a Message** - Not implemented in Netrexx Pipelines.

tcpclient	**Connect to a TCP/IP Server and Exchange Data**
	
	- Simple tcpclient implementation. The options implemented are similar to the CMS definition.
	  - linger - wait a bit before terminating the last read (units SECONDS)
	  - timeout - wait this long before timing reads out (units MS)
	  - deblock - If deblock is omitted a copy stage is used.
	  - group - similar to CMS. A delimited string containing a stage is expected. You can use a run of stages, but its is dangerous since you to know the stage sep character being used...
	  - greeting - expect a greeting message and discard it
	  - nodelay - use the nodelay option
	  - keepalive - enable keep alive socket option
	  - oneresponse - synchronize cmds/replys

tcpdata	**Read from and Write to a TCP/IP Socket**
	
	- Simple tcpdata implementation.
	  - linger - wait a bit before terminating the last read (units SECONDS)
	  - timeout - wait this long before timing reads out (units MS)
	  - deblock - If deblock is omitted a copy stage is used.
	  - group - similar to cms. A delimited string containing a stage is expected. You can use a run of stages, but its is dangerous since you need to know the stage sep character being used...
	  - nodelay - use the nodelay option
	  - oneresponse - synchronize requests/replies

tcplisten	**Listen on a TCP Port** - Simple tcplisten implementation. You can only supply the port and a timeout value, which is ignored unless tcplisten stream has been severed, in which case tcplisten terminates. - If input stream 0 is connected, tcplisten does a peekto before calling the accept method. The object is consumed af output of the socket object returns.

terminal
termina
termina
termin
termi
term
console
consol
conso
cons
cons
3.11
	Read or Write the Terminal in Line Mode
	
	``
	►►--+-TERMinal-+--+----------------------+--+-----------------------------+--| CMSonly
	     +-CONSole--+ +-EOF--delimitedString-+ +-PRfix--+-delimitedString-(1)+
	                    +-NOEOF----------------+ +-PRompt-+
	
	  CMSonly:
	    -|-+---------------+-(2)-|-
	       +-DIRECT---------+
	       +-ASYNchronously-+
	       +-DARK-----------+
	``
	
	- (1) NetRexx only
	  On first stage, delimitedString is put out as a prompt
	  On other stages, each line is prefixed with delimitedString
	  Outout to next stage does NOT include delimitedString
	  Either keyword can be used for either stage
	- (2) CMS only

threeway	**Split record three ways** - Not implemented in Netrexx Pipelines.

timestamp	**Prefix the Date and Time to Records**
	
	``
	                         (1) (2)
	►►--TIMEstamp--+--+----------------------+--+------►◄
	
			+-8------+	|
				
					+--+-number-+--+-------+--+ |
					
						+-number-+ |
						               +-SHOrtdate--------------------+ (3/09/46 23:59:59)
						               +-ISOdate----------------------+ (1946-03-09 23:59:59)
						               +-FULLdate---------------------+ (3/09/1946 23:59:59)
						               +-STAndard---------------------+ (19460309235959)
						               +-STRing--delimitedString--(3)---+
						``
						
						- (1) In CMS Pipelines, the delimited string is required. In NetRexx Pipelines, it defaults to // if no second string.

tokenise *tokenize*	**Tokenise Records** ``` ►►--+-TOKENISE-+--+--------------------+--+---------------+--►◄ +-TOKENIZE-+ +-delimitedString-(1)-+ +-delimitedString-+ ``` - (1) In CMS Pipelines, the first delimited string is required. In NetRexx Pipelines, it defaults to // if no second string.
tolabel *tolabe* *tolab*	**Select Records to the First One with Leading String** ``` ►►--TOLABel--+----------+----------►◄ +--string--+ ```
totarget	**Select Records to the First One Selected by Argument Stage** ``` ►►--TOTARGET----stage--+------------+-------►◄ +--operands--+ ```
trackblock	**Build Track Record** - Not implemented in Netrexx Pipelines.
trackdeblock	**Deblock Track** - Not implemented in Netrexx Pipelines.

trackread	**Read Full Tracks from ECKD Device** • Not implemented in Netrexx Pipelines.
tracksquish	**Squish Tracks** • Not implemented in Netrexx Pipelines.
trackverify	**Verify Track Format** • Not implemented in Netrexx Pipelines.
trackwrite	**Write Full Tracks to ECKD Device** • Not implemented in Netrexx Pipelines.
trackxpand	**Unsquish Tracks** • Not implemented in Netrexx Pipelines.

translate
translat
translate
transl
trans
xlate
	Transliterate Contents of Records
	
	<pre>▶▶--+--TRANSlate--+-+---------------------+-+-----------------+--▶
	     +--XLATE------+ +--<u>inputRange</u>-----------+ +-| default-table |-+
	                   | ╭-----------------+ |
	                   +--+-(--<u>inputRange</u>--)-+--+
	
	        ╭--------------------+
	  ▶--+-+----------------+--+------▶◀
	        +--<u>xrange</u>--<u>xrange</u>--+
	
	default-table:
	    |--+--UPper------------------------+------|
	       +--LOWer-----------------------+
	       +--INput-----------------------+
	    { +--OUTput----------------------+ }
	    { +--+--TO----+--+-----------+--<i>n</i>--+ }
	    { +--FROM--+ +--CODEPAGE--+ }
	    { }
	    { Not yet in Pipes for NetRexx }</pre>

trfread	**Read a Trace File** • Not implemented in Netrexx Pipelines.

truncate
truncat
trunca
trunc
chop
	Truncate the Record
	
	<pre> +--<i>80</i>----------------+
	▶▶--+-TRUNCate-+--+-------------------+--▶◀
	     +-CHOP-----+ +-<i>snumber</i>------------|
	                   +--| stringtarget |--+
	stringtarget:
	  |--+-----------+--+------------------------+-+-----+--| target |--|
	     +-ANYcase-----+ | +-BEFORE-+ | +-NOT-+
	     +-CASEANY-----+ +--+---------+--+--------+--+
	     +-CASEIGNORE--+ +-<i>snumber</i>-+ +-AFTER--+
	     +-IGNORECASE--+
	     +-CASELESS----+
	target:
	  |--+--<u>xrange</u>----------------------+--|
	     +--+-STRing--+--<u>delimitedString</u>-+
	         +--ANYof--+</pre>

tso	**Issue TSO Commands, Write Response to Pipeline** • Not implemented in Netrexx Pipelines.
udp	**Read and Write an UDP Port** • Not implemented in Netrexx Pipelines.

unique
uniqu
uniq
	Discard or Retain Duplicate Lines
	
	<pre> +--NOPAD------+
	►►--UNIQue--+---------+--+------------+--+-------------+------►
	             +--COUNT--+ +--PAD--<u>*xorc*</u>--+ +-ANYcase-----+
	                                           +-CASEANY-----+
	                                           +-CASEIGNORE--+
	                                           +-IGNORECASE--+
	                                           +-CASELESS----+
	
	                                 +--LAST------+
	 ►--+-----------------+--+------------+--------------------►◄
	       +--| uniqueRanges |-+ +--SINGLEs---+
	                             +--FIRST-----+
	                             +--MULTiple--+
	                             +--PAIRwise--+
	
	  uniqueRanges:
	     |--+--<u>*inputRange*</u>----------------------------+--|
	
		↵----------------------------+ |
		        +-- (--+--<u>*inputRange*</u>--+------------+--+--)--+
		                             +--NOPAD------+
		                             +--PAD--<u>*xorc*</u>--+</pre>

unpack	**Unpack a Packed File** • Not implemented in Netrexx Pipelines.
untab	**Replace Tabulate Characters with Blanks** • Not implemented in Netrexx Pipelines.
update	**Apply an Update File** • Not implemented in Netrexx Pipelines.
urldeblock	**Process Universal Resource Locator** • Not implemented in Netrexx Pipelines.
uro	**Write Unit Record Output** • Not implemented in Netrexx Pipelines.
utf	**Convert between UTF-8, UTF-16, and UTF-32** • Not implemented in Netrexx Pipelines.
var	**Retrieve or Set a Variable in a REXX or CLIST Variable Pool** <pre>►►--VAR--*variable*------------►◄</pre> • Pipes for NetRexx: this can only read vars
vardrop	**Drop Variables in a REXX Variable Pool** • Not implemented in Netrexx Pipelines.
varfetch	**Fetch Variables in a REXX or CLIST Variable Pool** • Not implemented in Netrexx Pipelines.

varload	**Set Variables in a REXX or CLIST Variable Pool** • Not implemented in Netrexx Pipelines.
varover 3.09	**Write the Values of Stems** ``` ┌NetRexx─────────────────┐ │ ►►--VAROVER--varName---►◄ │ └──────────────────────────┘ ``` • NetRexx Pipelines only; not CMS Pipelines
varset	**Set Variables in a REXX or CLIST Variable Pool** • Not implemented in Netrexx Pipelines.
vchar	**Recode Characters to Different Length** • Not implemented in Netrexx Pipelines.
vector	**Read or Write an Array of Vectors** • Pipes for NetRexx only.
vectora	**Add to an Array of Vectors** • Pipes for NetRexx only.
vectorr	**Read From an Array of Vectors** • Pipes for NetRexx only.
vectorw	**Write to an Array of Vectors** • Pipes for NetRexx only.

verify
3.09
	Verify that Record Contains only Specified Characters
	
	``
	                                        ┌──────────────────+ (1)
	►►--VERIFY--+-----------+-+-----------+-+-+-delimitedString-+-+--►◄
	            +-ANYCASE----+ +-inputRange-+ +-character-range-+ (1)
	            +-CASEANY----+
	            +-CASEIGNORE-+
	            +-IGNORECASE-+
	            +-CASELESS---+
	``
	
	• (1) NetRexx Pipelines only
	• (1) Examples: A-Z 0-9 c-g a4-ba; 16-bit Unicode characters or hex numbers
	• (1) Any number greater than zero, any order, of delimitdStrings and character-ranges are allowed.

vmc	**Write VMCF Reply** • Not implemented in Netrexx Pipelines.
vmcdata	**Receive, Reply, or Reject a Send or Send/receive Request** • Not implemented in Netrexx Pipelines.
vmclient	**Send VMCF Requests** • Not implemented in Netrexx Pipelines.
vmclisten	**Listen for VMCF Requests** • Not implemented in Netrexx Pipelines.
waitdev	**Wait for an Interrupt from a Device** • Not implemented in Netrexx Pipelines.

warp	**Pipeline Wormhole** • Not implemented in Netrexx Pipelines.
warplist	**List Wormholes** • Not implemented in Netrexx Pipelines.

whilelabel
3.09
	Select Run of Records with Leading String
	
	<pre>▶▶--WHILELABEL-+---------+---▶◀
	              +-*string*--+</pre>

wildcard	**Select Records Matching a Pattern** • Not implemented in Netrexx Pipelines.
writepds	**Store Members into a Partitioned Data Set** • Not implemented in Netrexx Pipelines.
xab	**Read or Write External Attribute Buffers** • Not implemented in Netrexx Pipelines.
xedit	**Read or Write a File in the XEDIT Ring** • Not implemented in Netrexx Pipelines.

xlate
translate
translat
transla
transl
trans
	Transliterate Contents of Records
	
	<pre>▶▶--+--XLATE------+--+----------------------+-+------------------+-▶
	    +--TRANSlate--+ +--<u>inputRange</u>-----------+ +-| default-table |-+
	                     | ┌-----------------+ |
	                     +--+-(--<u>inputRange</u>--)-+--+
	
	      ┌----------------------+
	 ▶--+--+-----------------+--+------▶◀
	         +--<u>xrange</u>--<u>xrange</u>--+
	
	default-table:
	    |--+--UPper------------------------+-------|
	       +--LOWer------------------------+
	       +--INput------------------------+
	    { +--OUTput-----------------------+ }
	    { +--+--TO----+--+-----------+--*n*--+ }
	    { +--FROM--+ +--CODEPAGE--+ }
	    { }
	    { Not yet in Pipes for NetRexx }</pre>

xmsg	**Issue XEDIT Messages** • Not implemented in Netrexx Pipelines.
xpndhi	**Expand Highlighting to Space between Words** • Not implemented in Netrexx Pipelines.

xrange
3.09
	Write a Range of Characters
	
	``
	►►--XRANGE--+------------+------►◄
	            +-xrange-----│
	            +-xorc--xorc-+
	``
	
	• NetRexx uses UTF-16 (ASCII) and CMS uses EBCDIC

zone	**Run Selection Stage on Subset of Input Record**
	
	``
	►►--ZONE--+------------------------------------+--►
	          +--+--WORDSEParator---+---+--char-----+--+
	             +--WS-------------+ +--hexchar--+
	             +--FIELDSEparator--+ +--BLANK----+
	             +--FS-------------+ +--SPACE----+
	
	 ►-+--Words----wNumberRange---+------►
	    +--Fields----fNumberRange--+
	    +--cNumberRange-----------+
	
	 ►--+---------+---+----------+--stage--+-----------+---►◄
	     +--CASEI--+ +--REVERSE--+ +--operands--+
	``