
Callable program modules

In addition to the utility commands, also included are some generic function program modules, which are used by NUTIL functions but can also be used by your own user programs.

These module programs are always identified by their program name, which always has an '@' as the first character of the program name. The module programs that can be used by your own programs are explained in this section of the manual.

The RPG routines all remain open after the first program call, in order to improve response times on program calls:

The OPM routines can be closed by the RCLRSC (Reclaim Resources) command.

The ILE routines can be closed by the RCLACTGRP (Reclaim Activation Group) command:

- Standalone ILE programs run in the QILE activation group
- Service programs run in the QSRVPGM activation group

Service Programs and the NUTIL Binding Directory

There are 2 service programs supplied with NUTIL:

- NSRVPGM is the general service program for all NUTIL callable modules except the 'Y2K Conversion Assistant' date conversion routines
- NDCSRV is the service program containing *only* the 'Y2K Conversion Assistant' date conversion modules

These service programs are also referenced in the NUTIL NBNDDIR binding directory. As a general rule you should always access the NUTIL service program modules via the NBNDDIR binding directory instead of directly accessing the service program. This ensures any changes to the structure/sequencing of the service programs (and their module imports/exports) will not affect the operation of any programs you have created that use them.

When creating your program you just specify the binding directory on your CRT command:

```
CRTPGM...    BNDDIR (NUTIL/NBNDDIR)
CRTBNDRPG... BNDDIR (NUTIL/NBNDDIR)
CRTBNDCL...  BNDDIR (NUTIL/NBNDDIR)
CRTBNDCBL... BNDDIR (NUTIL/NBNDDIR)
CRTBNDRPG... BNDDIR (NUTIL/NBNDDIR)
```

You should define the NUTIL binding directory after any of your own specific binding directory entries, to ensure the correct procedure is accessed by your program call.

All NUTIL Service Programs are set to run in activation group QSRVPGM.

They can be closed by the RCLACTGRP (Reclaim Activation Group) command

Prototypes and /COPY

Prototypes for using NUTIL callable functions can be found in source file NUTIL/PROTYPES. To use one of the pre-defined prototypes, just use the standard /COPY statement in your program:

```
/COPY NUTIL/PROTYPES,@RVSDAT8_H
```

Date Handling Overview

Date handling for 6 and 7 digit dates

NUTIL date processing routines for 6 and 7 digit dates use a full 7 digit date field, where the first digit in all instances is a century digit.

Year/Month/Day is Century/Year/Month/Day - CYMMDD
 Month/Day/Year is Century/Month/Day/Year - CMMDDYY
 Day/Month/Year is Century/Day/Month/Year - CDDMMYY

This allows you to handle the changeover to the 21st century (century digit is set to 0 or 1, where the 20th century = 0).

When a 6 digit date is passed to one of the utility programs, a 6 digit date will be returned (the century digit is ignored).

In general, the NUTIL date routines for 6 and 7 digit dates will only work on dates in the range 1940 – 2039

For a 7 digit date, where the century digit is passed as zero, the basic windowing rule will be applied - that years less than 40 are century 0 (19nn) and years greater than 39 are 1 (20nn). This is especially important to note for the @RVSDAT and @GETDAT modules.

Date handling for 8 digit dates

All NUTIL 8 digit date handling routines use an 8P0 parameter variable for passing dates and a 4A parameter variable for passing date type.

The date formats used by NUTIL date handlers are as follows:

<u>Format name</u>	<u>Description</u>	<u>Format</u>	<u>Data length</u>	<u>Formatted Length</u>	<u>Example</u>
*MDY	Month/Day/Year	mm/dd/yy	6	8	06/27/03
*DMY	Day/Month/Year	dd/mm/yy	6	8	27/06/03
*YMD	Year/Month/Day	yy/mm/dd	6	8	02/06/27
*CYMD	Century/Year/Month/Day	cyy/mm/dd	7	9	002/06/27
*JUL	Julian	yy/ddd	5	6	02/179
*LONGJUL	Extended Julian	yyyy/ddd	7	8	1902/179
*ISO	International Standards Organisation	yyyy-mm-dd	8	10	1999-06-27
*USA	IBM USA Standard	mm/dd/yyyy	8	10	06/27/1999
*EUR	IBM European Standard	dd.mm.yyyy	8	10	27.06.1999
*JIS	Japanese Industrial Standard Christian Era	yyyy-mm-dd	8	10	1999-06-27
*JOB	Current job format	See below	6	8	
*JB8	Full job format	See below	8	10	

With *YMD, *MDY and *DMY you can pass the routine a 6 digit date and the routine will validate it as though it were a date in the range 1940 - 2039.

It is important to understand that the *YMD, *MDY and *DMY formats will only work on dates in the range 1940 - 2039. For dates outside this range you must use another date format.

The special value of *JOB will assume the date format of the job running the date routine, which will resolve to either a *YMD, *MDY or *DMY (6 digit date) format. This is also known as *JOBRUN format.

The special value of *JB8 can be used in the @RVSDAT8 module to use the *JOBRUN format, but process a full 8 digit date.

@CHKDAT - Check date for validity

This program will test a given date for validity. If invalid, a value of 9999999 will be returned as the date. Note that the program remains open after the first call, to improve response.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Chkd7	7P0	I/O	Field containing the date
£Dfi7	1A	I	Date format in, either 'D' Day/Mth/Year 'M' Mth/Day/Year 'Y' Year/Mth/Day 'S' System format 'J' Job format

Where a 6 digit date is passed, a 6 digit date will be returned (i.e. the century digit is ignored). Note that the date passed must be valid date data within standard rules. A zero date is considered invalid.

The valid date range for this procedure is January 1, 1940, to December 31, 2039. Dates outside of this range will be treated as invalid dates.

@CHKDAT8 - Check 8 digit date for validity

This program will test a given date for validity. If the date passed to the routine is invalid, a value of 99999999 will be returned as the date. Note that the program remains open in the default activation group after the first call, to improve response.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Chkd8	8P0	I/O	Field containing the date
£Dfi8	4A	I	Date format in, either *DMY Day/Mth/Year *MDY Mth/Day/Year *YMD Year/Mth/Day *ISO System format *EUR EUR format *JIS JIS format *JUL JUL format *USA USA format *JOB Job format

Note that the date passed must be valid date data within standard rules. A zero date is considered invalid.

Example of usage

This code will test field ScreenDate to see whether it contains a valid date in the current *JOB format. If it fails the check then the program will send an 'Invalid date' message to the user.

```

* ...Check Date is valid (only if entered)...
C   ScreenDate   Ifne   0
C   ScreenDate   Andne   999999
C                   Call   '@CHKDAT8'
C                   Parm   ScreenDate   fChkd8
C                   Parm   '*JOB'       fDfi8

* ...Date does not conform to Job Format...
C   fChkd8       Ifeq   99999999
C                   Move   *ON           *IN98
C                   Move   'GEN0061'     Nmsgid
C                   Endif
C                   Endif

```

Example of Prototyped Call

```

/COPY NUTIL/PROTOTYPES,@CHKDAT8_H

C           CALLP      @CHKDAT8 (fChkd8:fDfi8)

* ...Date does not conform to Job Format...
C           If         fChkd8 = 99999999
C           Eval       *IN99 = *ON
C           Eval       NMsggid = 'GEN0008'
C           Endif
C           Endif

```

@CHKTIM - Check time for validity

This program will test a given time for validity. Also available as a bound call via the NSRVPGM service program.

If invalid, a value of 999999 will be returned as the time. Note that the program remains open after the first call, to improve response.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Chktm	6P0	I/O	Field containing the time

The time is assumed to be in hour/minute/second (*hhmmss*) format.

@CHKVN - Check name for validity

This program will test a given name to ensure it conforms to standard OS/400 object naming conventions. Refer to the IBM iSeries CL Reference Guide, under the topic '*Rules for specifying names*' for further information. Also available as a bound call via the NSRVPGM service program.

The validity check tests the following:

- The first character can be A-Z, or one of the allowed special characters for the installed character set
- The rest of the name can be A-Z, an allowed special character, 1-9, _ or .
- Minimum length of the name is 1.

If invalid, a value of 'Y' will be returned in the error flag parameter. Note that the program remains open after the first call, to improve response.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Chknm	10A	I	Field containing the name to be checked
£Nmerr	1A	O	Error flag. If invalid name, the 'Y' is returned

@CLCDAT – Calculate an 8 digit date, based on start date and duration

This program will calculate a date, given a base (starting) date and a duration to add or subtract from it. It can also return the day number or the week, the day name and the month name for the calculated date.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
Required parameters			
£Clcd8	8P0	I/O	Field containing the date
£Dfi8	4A	I	Date format in, either *DMY Day/Mth/Year *MDY Mth/Day/Year *YMD Year/Mth/Day *ISO System format *EUR EUR format *JIS JIS format *JUL JUL format *USA USA format *JOB Job format
£DurUnit	3P0	I	Number of units to add to, or subtract from, the input date
£DurType	8A	I	Duration type for the units specified, either *DAY *DAYS *MONTH *MONTHS *YEAR *YEARS
Optional parameters			
£Day	1P0	O	Day number of the week, where Monday is 1, to Sunday (7)
£DayName	10A	O	Day name (Monday, Tuesday, etc)
£MthName	10A	O	Month name (January, February etc)

Day and Month names text

Any “text descriptors”, like day and month names, are stored in the NUTIL/NUMSGF message file.

The message identifiers are NTX0001 to NTX0012 (month names) and NTX0013 to NTX0019 (day names).

If you wish to change these to display a language other than English, use the WRKMSGD (Work with Message Descriptions) command to change the text (maximum 10 characters for each message).

@CLCLOG - Calculate Logarithm

This program calculates the logarithm, either common (log to base 10) or natural (log to base e), based on an input number. Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Number	15P9	I/O	Field containing the input number. Also the field in which the calculated logarithm will be returned.
£LogType	1A	I	Calculation type, either '1' Common log '2' Natural log

If an error occurs in the calculation process, the number returned will be 0.

Note that the program remains open after the first call, to improve response.

@CMPDTS – Compare Date/Timestamps

This program will compare two timestamps and return the difference between the two, in seconds.
Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
Required parameters			
£Date1	8P0	I	Date 1, in *ISO format
£Time1	6P0	I	Time 1, in *HMS format
£Date2	8P0	I	Date 2, in *ISO format
£Time2	6P0	I	Time 2, in *HMS format
£Diff	11P0	O	Time difference, in seconds

It is assumed that, in normal operational circumstances, the Date1/Time1 timestamp will be *earlier than* the Date2/Time2 timestamp (see below).

Special notes:

- If any of the 4 input values are invalid, or if a program failure occurs, the value returned in output variable £Diff will be 9999999999
- If the Date1/Time1 timestamp is *greater (later) than* the Date2/Time2 timestamp, the value returned in output variable £Diff will be *negative*
- If the Date1/Time1 timestamp is *less (earlier) than* the Date2/Time2 timestamp, the value returned in output variable £Diff will be *positive*
- If the Date1/Time1 timestamp is *equal to* the Date2/Time2 timestamp, the value returned in output variable £Diff will be *zero*

The values in the input variables must be valid dates and times, in the format shown.

@CVTCDT - Convert 8 digit date from/to *CYMD format

This program will either convert an 8 digit date into *CYMD (CYY/MM/DD) format, or convert a *CYMD date into 8 digit date format, depending on the conversion type parameter passed. Also available as a bound call via the NSRVPGM service program.

The valid date range for this procedure is January 1, 1928, to December 31, 2071. Dates outside of this range will be treated as invalid dates.

The century digit in the *CYMD form of date is set to 0 for dates in the range 1928 through 1999 and is set to 1 for years 2000 through 2071.

If the expected input date is invalid, a value of 99999999 (or 9999999) will be returned as the output date.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Cvtd8	8P0	I/O	Date in *YYYYMMDD format
£Cvtd7	7P0	I/O	Date in *CYMMDD format
£CvtType	5A	I	Conversion type, either *FROM (from *CYMD to 8 digit date) or *TO (to *CYMD from 8 digit date)

@CVTJUL - Convert date from/to Julian format

This program will either convert a date from *CYMD format into Julian (*YYDDD) format, or convert a Julian date into *CYMD format, depending on the conversion type parameter passed. Also available as a bound call via the NSRVPGM service program.

The valid date range for this procedure is January 1, 1940, to December 31, 2039. Dates outside of this range will be treated as invalid dates.

If the input date is invalid, a value of 9999999 (or 99999) will be returned as the output date.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Cvtd7	7P0	I/O	Date in *CYMD format
£Cvtd5	5P0	I/O	Date in Julian format
£CvtType	5A	I	Conversion type, either *FROM (from Julian to *CYMD) or *TO (to Julian from *CYMD)

Where a 6 digit date is passed in the £Cvtd7 parameter, a 6 digit date will be returned (i.e. the century digit is ignored).

@CVTJUL8 - Convert 8 digit date from/to Julian format

This program will either convert a date from *YYYYMMDD format into 'long' Julian (*YYYYDDD) format, or convert a long Julian date into *YYYYMMDD format, depending on the conversion type parameter passed.

Also available as a bound call via the NSRVPGM service program.

The valid date range for this procedure is August 24, 1928, to May 9, 2071. Dates outside of this range will be treated as invalid dates.

If the input date is invalid, a value of 99999999 (or 9999999) will be returned as the output date.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Cvtd8	8P0	I/O	Date in *YYYYMMDD format
£Cvtd7	7P0	I/O	Date in Long Julian (*YYYYDDD) format
£CvtType	5A	I	Conversion type, either *FROM (from long Julian to *CYMD) or *TO (to long Julian from *CYMD)

@ExCmd - Process a command

This procedure allows the direct processing of a command from within an ILE program. The command string can be up to 2000 characters long, and a return code will advise whether the command was processed successfully or not.

Available only via the NSRVPGM service program.

Procedure Prototype

```
* Returns 0 if execution successful; otherwise 1.
D @ExCmd      PR          10I 0
D £Cmd        2000A
```

Usage from within ILE RPG program

```
* Return code from @ExCmd
D £Rc          10I 0

C              Eval      £Rc = @ExCmd('chgjob ' +
C                      %trim(JobNbr) + '/' +
C                      %trim(JobUsr) + '/' +
C                      %trim(JobNam) + ' OUTQ(' +
C                      %trim(Outq) + ')

C              If        £Rc = 0
*                      ... Command processed OK
C              Else
*                      ... £Rc = 1; Command processing failed
C              Endif
```

In this example, @ExCmd will execute the CHGJOB command and set the output queue to the name specified in the variable OUTQ.

If the execution of the command is successful, Variable £RC will return a value of 0; if it fails the value of £RC will be 1.

@ERRMSG - Error message handler

This program will perform a function on a program or external message queue, based on the input parameters. It can be used to

- Clear a program message queue
- Send a message to a program message queue
- Send a program status message

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£MsgID	7A	I	Message ID to send
£Msgf	10A	I	Message file containing MSGID
£Msgdta	120A	I	Message data to include in message

Special Usage of the @ERRMSG program

If the £MsgID parameter is set to the special value of blanks, the program message queue of the calling program will be cleared.

If the £MsgID parameter is set to the special value of *STATUS, a program status message will be displayed to the user, using the text passed in the £Msgdta parameter as the status message.

@GETDAT - Calculate a date, based on start date and no of days

This program will calculate a date, given a base (starting) date and a number of days to add or subtract from it. It can also return the day number or the week, the day name and the month name for the calculated date.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
Required parameters			
£Clcd8	7P0	I/O	Field containing the date
£Dfi	1A	I	Date format in, either 'D' Day/Mth/Year 'M' Mth/Day/Year 'Y' Year/Mth/Day 'S' System format 'J' Job format
£Days	3P0	I	Number of days to add to, or subtract from, the input date
£Day	1P0	O	Day number of the week, where Monday is 1, to Sunday (7)
Optional parameters			
£DayName	10A	O	Day name (Monday, Tuesday, etc)
£MthName	10A	O	Month name (January, February etc)

Where a 6 digit date is passed, a 6 digit date will be returned (i.e. the century digit is ignored).

Notes on 7 digit dates

The century portion of a 7digit date is implied, as follows:

- Years in the range 40-99 will be treated as 1940-1999
- Years in the range 00-39 will be treated as 2000-2039
- Any dates outside of the range 1940-2039 are invalid dates

Examples of usage:

If the program was passed:

```
CALL @GETDAT PARM(120488 D 28 0)
```

Then the date returned would be 100588.

And if the program was passed:

```
CALL @GETDAT PARM(120488 D -5 0)
```

Then the date returned would be 070488.

In the case of century changeover, if the date passed was

```
CALL @GETDAT PARM(0291299 D 5 0)
```

Then the date returned would be 1030100 (3rd January, 2000).

@GETDAT8 - Calculate an 8 digit date, based on start date and no of days

This program will calculate a date, given a base (starting) date and a number of days to add or subtract from it. It can also return the day number or the week, the day name and the month name for the calculated date.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
Required parameters			
£Clcd8	8P0	I/O	Field containing the date
£Dfi8	4A	I	Date format in, either *DMY Day/Mth/Year *MDY Mth/Day/Year *YMD Year/Mth/Day *ISO System format *EUR EUR format *JIS JIS format *JUL JUL format *USA USA format *JOB Job format
£Days	3P0	I	Number of days to add to, or subtract from, the input date
£Day	1P0	O	Day number of the week, where Monday is 1, to Sunday (7)
Optional parameters			
£DayName	10A	O	Day name (Monday, Tuesday, etc)
£MthName	10A	O	Month name (January, February etc)

@GETDAYS - Get the number of days between two dates

This program will calculate the number of days difference between the two dates supplied. Invalid dates will be treated as 0000000.

Also available as a bound call via the NSRVPGM service program.

The valid date range for this procedure is January 1, 1940, to December 31, 2039. Dates outside of this range will be treated as invalid dates.

Where an error in dates occurs, the number of days parameter will return 999999 to the calling program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Date1	7P0	I	'From' Date in *CYMD format
£Date2	7P0	I	'To' Date in *CYMD format
£Days	6P0	O	The number of days calculated between the dates

Where a 6 digit date is passed, a 6 digit date will be returned (i.e. the century digit is ignored).

@GETDYS8 - Get the number of days between two 8 digit dates

This program will calculate the number of days difference between the two dates supplied. Invalid dates will be treated as 00000000.

Also available as a bound call via the NSRVPGM service program.

The valid date range for this procedure is August 24, 1928, to May 9, 2071. Dates outside of this range will be treated as invalid dates.

Where an error in dates occurs, the number of days parameter will return 999999 to the calling program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Date1	8P0	I	'From' Date, in *YYYYMMDD format
£Date2	8P0	I	'To' Date, in *YYYYMMDD format
£Days	6P0	O	The number of days calculated between the dates

@GETDOW - Get the day-of-the-week number for a given date

This program will determine the day-of-the-week number for a given date. Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Date8	8P0	I	Field containing the date, in *YYYYMMDD format
£Day	1P0	O	Day number of the week, where 1=Monday, though to 7=Sunday

If the date passed to the routine is invalid for the format specified, a value of 9 will be returned as the day number.

@GETTIM - Calculate time

This program will calculate a time, based on a start time and a duration to add or subtract. Also available as a bound call via the NSRVPGM service program.

If invalid, a value of 999999 will be returned as the time. Note that the program remains open after the first call, to improve response.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Time	6P0	I/O	Field containing the time
£Hours	5P0	I	Field containing the hours duration to apply
£Minutes	5P0	I	Field containing the minutes duration to apply
£Seconds	5P0	I	Field containing the seconds duration to apply

The input time is assumed to be in hour/minute/second (*hhmmss*) format.

Depending on the values in each of the duration parameters, the hours duration will be applied to the time, then the minutes and finally the seconds.

No error is returned if the calculated time 'wraps' to a future or previous date.

@GETWKNR - Get the ISO week number for a given date

This program will determine the week number-of-the-year for a given date. The week number is based on the International Standards Organisation standard for week number definition.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Date7	7P0	I	Field containing the date, in *CYMD format
£Weeknr	2P0	O	ISO week number of the year

Date can be passed as either a 6 digit number (YYMMDD) or a 7 digit number (CYMMDD). If a 6 digit number is passed, the century digit will be assumed to be 0.

If the date passed to the routine is invalid, a value of 0 will be returned as the week number.

@GETWKN8 - Get the ISO week number for a given date

This program will determine the week number-of-the-year for a given date. The week number is based on the International Standards Organisation standard for week number definition.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Date8	8P0	I	Field containing the date, in *YYYYMMDD format
£Weeknr	2P0	O	ISO week number of the year

If the date passed to the routine is invalid, a value of 0 will be returned as the week number.

@RANDOM - Generate a Random Number

This program generates random numbers.

Also available as a bound call via the NSRVPGM service program.

The first call to the program generates a 'seed' number based on the current date/time. The initial seed number is based on the formula:

```
f(z) = az mod m
           where a = 16807
           m = 2147483647
```

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Loval	10P0	I	Lower limit of range
£Hival	10P0	I	Upper limit of range
£Rndn	10P0	O	Random number

The lower limit and upper limit parameters allow you to specify the limits between which the random number can be generated. So the following call:

```
CALL @RANDOM PARM(15 45 0)
```

would generate a random number between 15 and 45.

Note that the program remains open after the first call, to improve response.

@RTVASI - Retrieve Auxiliary Storage information

This program will retrieve Auxiliary Storage information relating to disk usage of the system.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£SysAsp	7P0	O	Storage capacity of System ASP 1 (in Mb)
£SysAspC	7P4	O	Percentage of System ASP currently used
£TotalAsp	7P0	O	Total auxiliary storage on system (in Mb)
£CurUnpUse	7P0	O	Current unprotected storage used (in Mb)
£MaxUnpUse	7P0	O	Maximum unprotected storage used (in Mb)

@RTVDATE - Retrieve current system date/time stamp

This program will retrieve the current system date and time and return the date in the format specified.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Dfo	1A	I	Date format in, either 'D' Day/Mth/Year 'M' Mth/Day/Year 'Y' Year/Mth/Day 'S' System format 'J' Job format
£Sysd7	7P0	O	Field containing the system date
£System	6P0	O	Field containing the time

Where a 6 digit date parameter is passed, a 6 digit date will be returned (i.e. the century digit is ignored).

Notes on 7 digit dates

The century portion of a 7digit date is implied, as follows:

- Years in the range 40-99 will be treated as 1940-1999
- Years in the range 00-39 will be treated as 2000-2039
- Any dates outside of the range 1940-2039 are invalid dates

@RTVDATE8 - Retrieve current 8 digit system date/time stamp

This program will retrieve the current system date and time and return the date in *ISO format.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Sysd8	8P0	O	Field containing the system date in YYYYMMDD format
£System	6P0	O	Field containing the time

@RTVDTI - Retrieve date information

This program will return the following information about a given date:

- The day number of the week (where Monday is 1, Sunday is 7)
- The day number of the year
- The ISO week number of the year

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Date	7P0	I	Field containing the date in *CYMD format
£DayOfWeek	1P0	O	Day number of the week
£DayOfYear	3P0	O	Day number of the year
£WeekOfYr	2P0	O	Week number of the year

Date can be passed as either a 6 digit number (YYMMDD) or a 7 digit number (CYMMDD). If a 6 digit number is passed, the century digit will be assumed to be 0.

@RTVDTS - Retrieve *ISO date/time stamp

This program will retrieve the current system date and time and return the date in *ISO format. Retrieved time includes the milliseconds portion of the timestamp.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Isodt	8P0	O	Field containing the system date in YYYYMMDD format
£System9	9P0	O	Field containing the time in HHMMSSsss

The result is the same as a CALL to the @RTVDAT8 program, but includes milliseconds in the time parameter.

@RTVDFM - Retrieve system/job date formats

This program will retrieve the values for the system date format and the current job date format.
Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£SysFmt1	1A	O	System date format, where Y=Year/Month/Day M=Month/Day/Year D=Day/Month/Year
£JobFmt1	1A	O	Current job date format, where Y=Year/Month/Day M=Month/Day/Year D=Day/Month/Year

@RTVDF8 - Retrieve 8 digit system/job date formats

This program will retrieve the values for the system date format and the current job date format.
Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£SysFmt	4A	O	System date format, where *YMD=Year/Month/Day *MDY=Month/Day/Year *DMY=Day/Month/Year
£JobFmt	4A	O	Current job date format, where *YMD=Year/Month/Day *MDY=Month/Day/Year *DMY=Day/Month/Year

@RTVEMLADR - Retrieve SMTP Email Address

This program will retrieve the SMTP email address for a specified user. Information is retrieved from the system directory.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£UserID	8A	I	Field containing the User ID in the System Distribution Directory
£UserAddr	8A	I	Field containing the User Address in the System Distribution Directory
£EmailAddr	256A	O	Field containing the user's SMTP email address

If the specified user does not have an SMTP email address defined in the System Distribution Directory, the value returned will be blanks.

@RTVIPA - Retrieve the Session IP address

This program will retrieve the TCP/IP address for the current signed-on workstation session.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£IPAddress	15A	O	Field containing the IP address in dotted decimal form <i>nnn.nnn.nnn.nnn</i>

If the current job is not a workstation session connected via TCP/IP, the value returned will be blanks.

@RTVJDT8 - Retrieve current 8 digit job date/time stamp

This program will retrieve the current job date and time. Date is returned in *ISO format.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Jobd8	8P0	O	Field containing the job date in YYYYMMDD format
£JobTm	6P0	O	Field containing the time

@RTVMSG - Retrieve Message

The Retrieve Message API is used to retrieve a specified predefined message from a message file and to copy the First Level Text in to a returned variable.

Substitution values can be specified in the £MSGDTA parameter (as a single character string containing one or more concatenated message data fields) to replace the substitution variables in the predefined message text.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Msgid	7A	I	Message ID to retrieve
£Msgf	10A	I	Message file containing the Message ID
£MsgDta	80A	I	Message data to include in the returned message
£Msg	80A	O	Returned message text

£MSGID specifies the message identifier of the predefined message that is being retrieved from the specified message file. The message ID must already exist in the message file specified.

£MSGF specifies the name of the message file containing the predefined message. The user library list is used to determine the location of the message file; all libraries in the job's library list are searched until the first match is found.

£MSGDTA specifies the substitution values that are used in the retrieved message if the predefined message contains substitution variables. If more than one substitution variable exists, the message data is concatenated in to this parameter; the formatting of the concatenated substitution data in to the message text is then performed based on the message data definition in the predefined message.

£MSG specifies the name of the character variable in the program into which the First Level Text of the retrieved message is copied.

@RVSDAT - Reformat date to a new format

This program will reformat a given date, from one date format to another. The input date will then be returned to the requesting program in the requested format.

Also available as a bound call via the NSRVPGM service program.

If the date is invalid, a value of 9999999 will be returned as the date. If the date passed to the program is either 0 or 999999, it will be ignored by the program. Note that the program remains open after the first call, to improve response.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Sysd7	7P0	I/O	Field containing the date
£Dfi7	1A	I	Date format in, either 'D' Day/Mth/Year 'M' Mth/Day/Year 'Y' Year/Mth/Day 'S' System format 'J' Job format
£Dfo7	1A	I	Date format out, either 'D' Day/Mth/Year 'M' Mth/Day/Year 'Y' Year/Mth/Day 'S' System format 'J' Job format

Where a 6 digit date is passed, a 6 digit date will be returned (i.e. the century digit is ignored).

Notes on 7 digit dates

When not passed as input the century portion of a 7digit date is implied, as follows:

- Years in the range 40-99 will be treated as 1940-1999
- Years in the range 00-39 will be treated as 2000-2039
- 6 digit dates outside of the range 1940-2039 are invalid dates

@RVSDAT8 - Reformat 8 digit date to a new format

This program will reformat a given date, from one date format to another. The input date will then be returned to the requesting program in the requested format.

Also available as a bound call via the NSRVPGM service program.

If the date passed to the routine is invalid for the format specified, a value of 99999999 will be returned as the date.

A date of 0 is an invalid date.

A date of 99999999 will be ignored by the program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Isodt	8P0	I/O	Field containing the date
£Dfi	4A	I	Date format in, either *DMY Day/Mth/Year *MDY Mth/Day/Year *YMD Year/Mth/Day *ISO System format *EUR EUR format *JIS JIS format *JUL JUL format *USA USA format *JOB Job format (6 digits) *JB8 Job format (8 digits)
£Dfo	4A	I	Date format out, either *DMY Day/Mth/Year *MDY Mth/Day/Year *YMD Year/Mth/Day *ISO System format *EUR EUR format *JIS JIS format *JUL JUL format *USA USA format *JOB Job format (6 digits) *JB8 Job format (8 digits)

Notes on 8 digit dates

The century portion of the date is implied for date formats *DMY, *MDY and *YMD, as follows:

- Years in the range 40-99 will be treated as 1940-1999
- Years in the range 00-39 will be treated as 2000-2039
- 6 digit dates outside of the range 1940-2039 are invalid dates

The century portion of the date is a required entry for date formats *ISO, *USA, *EUR, *JIS. If not entered, the century portion will be treated as 00.

Example of usage

This code will convert the contents of the FileDate field from *ISO to *JOB format and move the result into the field ScreenDate.

```
* ...Reverse date from *ISO format to *JOB format...
C   FileDate      Ifne      0
C   FileDate      Andne     99999999
C   Call          '@RVSDAT8'
C   ScreenDate    Parm      FileDate      @Date8
C   Parm          '*ISO'      @Dfi8
C   Parm          '*JOB'      @Dfo8
C   Else
C   Z-add        0           ScreenDate
C   Endif
```

@RTVOSL - Retrieve the current i5/OS Operating System version

This program will retrieve the i5/OS Operating System version installed.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Version	6A	O	Field containing the release level in the form <i>VnRnMn</i>

@RTVPLI - Retrieve the Operating System primary language identifier

This program will retrieve the primary language load identifier for the current i5/OS Operating System version installed.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£LangID	4A	O	Field containing the primary language identifier.

A list of the valid language identifier codes can be found in the IBM i5/OS Globalization (National Language Support) Guide.

A code of '2924' indicates that English is the primary language installed.

@RTVRC – Retrieve current Record Count for specified file

This program will return the current number of non-deleted records in the specified file. If the specified file or member is not found, the returned record count will be zero.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£File	10A	I	Field containing the File Name
£Lib	10A	I	Field containing the Library Name
£Mbr	10A	I	Field containing the File Member Name
£RcdCnt	10P0	O	Return field containing the number of current records

Special values allowed for £Lib Library Name parameter:

*LIBL
*CURLIB

Special values allowed for £Mbr Member Name parameter:

*FIRST
*LAST

@RTVRL – Retrieve Record Length of specified file

This program will return the record length of the specified file. If the specified file is not found, the returned record length will be zero.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£File	10A	I	Field containing the File Name
£Lib	10A	I	Field containing the Library Name
£RcdLen	5P0	O	Return field containing the file record length

Special values allowed for £Lib Library Name parameter:

*LIBL
*CURLIB

@RTVSYSA - Retrieve system attributes

This program will retrieve information relating to the iSeries system (or partition) that the current job is running on.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Sysnm	8A	O	Field containing the System Name
£Srlnbr	8A	O	Field containing the System Serial number
£Model	4A	O	Field containing the System Model number

@RTSVI - Retrieve server information

This program will retrieve information about the currently configured iSeries NetServer (iSeries Support for Windows Network).

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£SvrNm	15A	O	Field containing the Server Name
£SvrNmp	15A	O	Field containing the pending Server Name
£DomNm	15A	O	Field containing the Domain Name
£DomNmp	15A	O	Field containing the pending Domain Name
£Text	50A	O	Field containing the Text Description

@RTVTIM - Retrieve current system time

This program will retrieve the current system time in the format requested.

Also available as a bound call via the NSRVPGM service program.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£TimeFmt	1A	O	Time format requested, where S=System time G=GMT
£Time	6P0	O	Field containing the system time in HHMMSS format

The return of GMT Time (or *Coordinated Universal Time, UTC*) is dependent on the correct setting of the system value QUTCFFSET.

QUTCFFSET is the system value containing the coordinated universal time offset, which is a value that defines the time difference between the local time and UTC (also known as Greenwich Mean Time, or Zulu). Refer to the IBM iSeries Work Management guide for further details.

If GMT time was requested and QUTCFFSET is not set, the program will return 999999 as the time.

If GMT time was requested and the system country code is GB (United Kingdom), the program will assume the system time is being requested.

@SNDSTS - Send a status message

This program will format a message and send it as a status message.

It is mainly supplied for use by RPG programs, due to the language restriction that they cannot directly send a status message.

Parameter List

<u>Parameter</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
£Msgid	7A	I	Message ID to send
£Msgf	10A	I	Message file containing MSGID
£StsDta	120A	I	Status Message data to include in message

If the £Msgid parameter is blank, the standard IBM message ID CPF9898 will be assumed and the contents of the message data parameter (£StsDta) will be sent as the status message.

The status message data (£StsDta) is used in one of two ways; either it contains the message to be sent (£Msgid is blank) or it contains variable data that is to be loaded into the message specified (£Msgid) before it is sent as a message.

Date 'Conversion Assistant' modules

The following NUTIL 'Date Conversion Assistant' programs can be used for manipulating your date data between 6 and 8 digit date formats.

Conversion Module	Purpose
@DCDDMMYY	Convert DDMMYY (day/month/year) format to DDMMYYYY
@DMDDYY	Convert MMDDYY (month/day/year) format to MMDDYYYY
@DCYY	Convert YY (year) format to YYYY
@DCYYDDD	Convert YYDDD (julian) format to YYYYDDD (longjul)
@DCYYMMDD	Convert YYMMDD (year/month/day) format to YYYYMMDD
@DCYYP	Convert YYPP (year/period) format to YYYYPP
@DCYYPWW	Convert YYPPWW (year/period/week) format to YYYYPPWW
@DCYYPWWD	Convert YYPPWWD (year/period/week/day) format to YYYYPPWWD
@DCYYWW	Convert YYWW (year/week) format to YYYYWW
@DCYYWWD	Convert YYWWD (year/week/day) format to YYYYWWD

In all cases of the above, the following is true:

- The call has two parameters, *DateVarIn* (date variable input) and *DateVarOut* (date variable output).
- Where the value of *DateVarIn* is either 0 or all 9's it will be ignored by the conversion (not @DCYY)
- No value judgement is made on the input data; it is assumed that you are passing the conversion module valid date data in the form expected.
- Years in the range 28-99 will be converted to 1928-1999
- Years in the range 00-27 will be converted to 2000-2027
- The program remains open after the first call, to improve response.
- The program can either be called directly, or as a bound call via the NUTIL NDCSRV service program (which can be assessed from the NBNDDIR binding directory).

@DCDDMMYY - Convert DDMMYY to DDMMYYYY

This program will convert a Day/Month/Year date variable into full 4 digit year representation.

Parameter List

<u>Field</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
@DdMmYy	6P0	I	DateVarIn, in DDMMYY format
@DdMmYyyy	8P0	O	DateVarOut, in DDMMYYYY format

@DCMMDDYY - Convert MMDDYY to MMDDYYYY

This program will convert a Month/Day/Year date variable into full 4 digit year representation.

Parameter List

<u>Field</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
@MmDdYy	6P0	I	DateVarIn, in MMDDYY format
@MmDdYyyy	8P0	O	DateVarOut, in MMDDYYYY format

@DCYY - Convert YY to YYYY

This program will convert a Year date variable into full 4 digit year representation.

Parameter List

<u>Field</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
@Yy	2P0	I	DateVarIn, in YY format
@Yyyy	4P0	O	DateVarOut, in YYYY format

@DCYYDDD - Convert YYDDD to YYYYDDD

This program will convert a Julian date variable into full 4 digit year representation (LongJulian).

Parameter List

<u>Field</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
@YyDdd	5P0	I	DateVarIn, in YYDDD format
@YyyyDdd	7P0	O	DateVarOut, in YYYYDDD format

@DCYMMDD - Convert YMMDD to YYYYMMDD

This program will convert a Year/Month/Day date variable into full 4 digit year representation.

Parameter List

<u>Field</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
@YyMmDd	6P0	I	DateVarIn, in YMMDD format
@YyyyMmDd	8P0	O	DateVarOut, in YYYYMMDD format

@DCYPP - Convert YPP to YYYYPP

This program will convert a Year/Period date variable into full 4 digit year representation.

Parameter List

<u>Field</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
@YyPp	4P0	I	DateVarIn, in YPP format
@YyyyPp	6P0	O	DateVarOut, in YYYYPP format

@DCYPPWW - Convert YPPWW to YYYYPPWW

This program will convert a Year/Period/Week date variable into full 4 digit year representation.

Parameter List

<u>Field</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
@YyPpWw	6P0	I	DateVarIn, in YPPWW format
@YyyyPpWw	8P0	O	DateVarOut, in YYYYPPWW format

@DCYPPWWD - Convert YPPWWD to YYYYPPWWD

This program will convert a Year/Period/Week/DayOfTheWeek date variable into full 4 digit year representation.

Parameter List

<u>Field</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
@YyPpWwD	7P0	I	DateVarIn, in YPPWWD format
@YyyyPpWwD	9P0	O	DateVarOut, in YYYYPPWWD format

@DCYYWW - Convert YYWW to YYYYWW

This program will convert a Year/Week date variable into full 4 digit year representation.

Parameter List

<u>Field</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
@YyWw	4P0	I	DateVarIn, in YYWW format
@YyyyWw	6P0	O	DateVarOut, in YYYYWW format

@DCYYWWD - Convert YYWWD to YYYYWWD

This program will convert a Year/Week/DayOfTheWeek date variable into full 4 digit year representation.

Parameter List

<u>Field</u>	<u>Defn</u>	<u>Usage</u>	<u>Description</u>
@YyWwD	5P0	I	DateVarIn, in YYWWD format
@YyyyWwD	7P0	O	DateVarOut, in YYYYWWD format