GXSHX$ - Execute Windows Shell Command on GX

The GXSHX$ routine can be used to execute a Windows Shell command on the PC that is running the current GX session.

1. Invocation

To execute a Windows Shell command on the PC that is running GX code:

```
CALL GXSHX$ USING filename [wait]
```

where *filename* is a zero-terminated string that specifies the "document file" that the Windows Shell command is to operate on; and *wait* is an optional PIC 9(4) COMP field, or literal, that specifies whether GX is to wait for the Windows application to complete:

0	Do not wait for program to exit (i.e. return control immediately)
1	Wait for program to exit

2. STOP Codes and Exception Conditions

The following STOP codes may be generated by GXSHX$:

STOP code	Description
23901	GXSHX$ has been called by an application that is not running on GX.
23902	Unknown version of SH control block (i.e. SHVERS does not contain 1)
23903	GX command buffer overflow
23904	Incompatible version of GX
23905	A document filename was not specified

The following exception conditions may be returned by GXSHX$:

EXIT code	$$COND	Description
23911	11	Unable to allocate data page for GX command buffer.

23912	12	The Windows ShellExecute operation failed. The Windows 32-bit error code is available in the $$CRES System Variable.
23913	13	The replaceable file parameter is not present in the GX.INI file.

3. Programming Notes

GXSHX$ is only available when running on GX. Any attempt to use GXSHX$ on a non-GX terminal will result in a STOP code.

The filename string MUST be terminated by a byte of binary-zero (i.e. #00, LOW-VALUES).

Unlike the SHCMD$ `shcmd$` or GXCMD$ `shcmd$` routines the GXSHX$ command does not require the name of the executable program to be supplied. The parameter supplied to GXSHX$ should be the name of a "document file" (e.g. *.txt, *.doc, *.pdf, *.htm etc.). The Windows File Association between the supplied "document file" and the associated executable will be used to execute the appropriate command. See section 4 for some examples.

By default, unless the *wait* parameter is specified, control will be returned to the calling application immediately.

The GXSHX$ interface is as simple as possible (i.e. only the name of the shell command or "document" file is required). The GXSHL$ `gxshl$` routine, which involves a more complex calling convention, allows more control over the execution of the shell command.

4. Examples

The following example invokes Word to open document EXAMPLE.DOC:

```
DATA DIVISION
77   X-DOC       PIC X(?)
     VALUE       "EXAMPLE.DOC"
     VALUE       #00

PROCEDURE DIVISION

     CALL GXSHX$ USING X-DOC
```

The following example invokes the standard browser open document EXAMPLE.HTM:

```
DATA DIVISION
77   X-HTM       PIC X(?)
     VALUE       "EXAMPLE.HTM"
     VALUE       #00

PROCEDURE DIVISION

     CALL GXSHX$ USING X-HTM
```

5. Copy-Books

None.

6. See Also

GXSHL$ - Execute Windows shell command on GX.
SHCMD$ - Execute Windows application
GXCMD$ - Execute Windows application on GX