## Identification

branch
R. J. Feiertag, S. Rosenbaum

## Purpose

This command creates a branch with a length of zero.

## Usage

At command level:

        branch  path

As a subroutine:

        call branch_(path, code);

    dcl  path char(*),              /*path pointing to
                                    branch to be added*/

         code fixed bin(17);        /*command system
                                    error code*/

If path is specified as a directory (i.e., ends with ">"),
a directory branch is created with the name of the entry
effectively pointed to by path and in the directory effectively
pointed to by path.  If path is not specified as a directory,
a non_directory branch with a zero length segment is created.
The maximum length of the segment is set to 64 1024-word
blocks.  The entry effectively pointed to must not exist.
Append mode is necessary in the directory of the entry
effectively pointed to.

## Examples

1)   branch  >user_dir_dir>fred>

        The directory "fred" is created in the directory
        ">user_dir_dir".

2)   branch (newseg1  newseg2)

        The branches "newseg1" and "newseg2" are created
        in the working directory.

3)   If the link "apple" in the working directory effectively
     points to the non-existent entry  ">user_dir_dir>fruits>pear"
     then the command:

        branch apple

    will create a branch "pear" in the directory
    ">user_dir_dir>fruits".

Implementation

setpath breaks the path name into a directory and an entry
name.  ufo$chase (BY.2.01) is called to find the nonexistent
branch pointed to and this branch is created by a call
to ufo$appendb (BY.2.01).  Errors are reported by a call
to command_error.