Assignment 3 Client Design

Wilson Sue
October 18, 2023

Structures

State

Field	Purpose
MAX_ARGS	Limits the maximum arguments given to 10.
MAX_PATH_LENGTH	Limits the maximum path to 256.
MAX_DIRECTORIES	**Limits the maximum directories to 256.**
CMD_FIFO	The path to the command FIFO.
OUT_FIFO	The path to the output FIFO (constructed from CMD_FIFO).
buffer	A buffer to read data from the output FIFO.
bytesRead	The number of bytes read from the FIFO in one iteration.

Command

main	The primary function where the execution begins.
strcmp	To compare two strings. Used to validate the -fifo argument.
write	To write data to a file descriptor (like STDOUT_FILENO or a FIFO).
argv	The arguments passed to the command
strlen	To compute the length of a string.
open	To open a file or FIFO for reading or writing.
snprintf	To format and store a string in the specified buffer.
read	To read data from a file descriptor (like a FIFO).
close	To close an open file descriptor.
dup2	To redirect one file descriptor to another.

Finite State Machine

State Table

From State	To State	Action
SERVER_INIT	SETUP_FIFOS	setup_fifo_names
SETUP_FIFOS	READ_CMD	open_and_read_command_fifo
READ_CMD	EXECUTE_CMD	fork_and_execute_command
EXECUTE_CMD	READ_CMD	continue_reading_next_command
*	ERROR_STATE	handle_error
ERROR_STATE	SERVER_EXIT	exit_server

Functions

main

Purpose

The server.c program acts as a server-side handler for command execution. It listens on a named FIFO for incoming command requests from a client. Upon receiving a command, it executes the command and sends the output back to the client through another FIFO.

Return

Type	Next State
Success	`Sends command to server and reads output from server`
Failure	`ERROR`

Pseudocode

```
INITIALIZE required constants and variables
PARSE command-line arguments

WHILE true:
    OPEN command FIFO for reading
    READ command from FIFO
    IF error reading:
        HANDLE error and continue

    PARSE received command into arguments
```

```
    FORK a new process to handle the command
    IF child process:
        REDIRECT stdout to output FIFO
        GET PATH variable to find directories for command
execution
        PARSE PATH variable into directories
        EXECUTE command from found directory
        IF command not found in any directory:
            DISPLAY error
    END IF
    WAIT for child to finish

    CLOSE command and output FIFOs
END WHILE

CLEANUP and exit
```