

的 shell 有大量的命令，比如 `pwd`、`jobs` 和 `fg`。

如果 `builtin_command` 返回 0，那么 shell 创建一个子进程，并在子进程中执行所请求的程序。如果用户要求在后台运行该程序，那么 shell 返回到循环的顶部，等待下一个命令行。否则，shell 使用 `waitpid` 函数等待作业终止。当作业终止时，shell 就开始新一轮迭代。

— *code/ecf/shell.c*

```

1  /* eval - Evaluate a command line */
2  void eval(char *cmdline)
3  {
4      char *argv[MAXARGS]; /* Argument list execve() */
5      char buf[MAXLINE];   /* Holds modified command line */
6      int bg;              /* Should the job run in bg or fg? */
7      pid_t pid;           /* Process id */
8
9      strcpy(buf, cmdline);
10     bg = parseline(buf, argv);
11     if (argv[0] == NULL)
12         return; /* Ignore empty lines */
13
14     if (!builtin_command(argv)) {
15         if ((pid = Fork()) == 0) { /* Child runs user job */
16             if (execve(argv[0], argv, environ) < 0) {
17                 printf("%s: Command not found.\n", argv[0]);
18                 exit(0);
19             }
20         }
21
22         /* Parent waits for foreground job to terminate */
23         if (!bg) {
24             int status;
25             if (waitpid(pid, &status, 0) < 0)
26                 unix_error("waitfg: waitpid error");
27         }
28         else
29             printf("%d %s", pid, cmdline);
30     }
31     return;
32 }
33
34 /* If first arg is a builtin command, run it and return true */
35 int builtin_command(char **argv)
36 {
37     if (!strcmp(argv[0], "quit")) /* quit command */
38         exit(0);
39     if (!strcmp(argv[0], "&")) /* Ignore singleton & */
40         return 1;
41     return 0; /* Not a builtin command */
42 }

```

— *code/ecf/shell.c*

图 8-24 eval 对 shell 命令行求值