

While the 'filenames' in this example do signal the function of the arguments, they are far from typical examples of filenames. Since real filenames do not typically refer to functions in rules, using this type of example may ultimately confuse the learner. The filenames are poor illustrations of what the 'fillers' of the argument slots may look like. The following situational example is better:

Suppose you have a file called BUDGET that contains your budget for 1986. Now you want to create a new budget for 1987, but you need a way to keep the files for the two years distinct. The command: (17)

```
RENAME BUDGET BUDGET.86
```

changes the name of the existing file BUDGET to BUDGET.86. Now you can create a file for the new budget called BUDGET.87, and it will be easy to distinguish the two files.

In addition to clarifying aspects of notation, this example also clarifies the functions of the two ordered arguments or parameters in the rule: the first is the old name and the second is the new name. The example also helps to motivate the use of the RENAME command, by presenting a situation in which renaming a file makes sense. As mentioned above, additional examples of the same sort may be needed to illustrate other aspects of rule.<sup>5</sup>

The results of Reder, Charney and Morgan (in press) suggest that rich examples of correct commands help people learn to generate their own commands. Indeed, elaborations on the execution of procedures proved to be more important to learners than elaborations on the function and motivation of the commands. We systematically varied whether or not a computer manual contained *syntactic* elaborations (e.g. examples of syntactically correct commands to illustrate more abstract rules for the commands) or *conceptual* elaborations (e.g. analogies illustrating the basic concepts, examples of situations in which a command would be useful). Factorially combining the two types of elaborations produced four versions of the manual. Figures 1 and 2 are corresponding excerpts from two of the manuals, describing the CHDIR ('Change Directory') command; Figure 1 contains just conceptual elaborations and Figure 2 contains just syntactic elaborations.

<sup>5</sup>The rule itself, taken from the DOS manual, is not very informative about the function of the arguments (or parameters). The following statement of the rule might be better:

```
RENAME [location and current name of file] [new name of file]
```

Research suggests, however, that even this form of the rule benefits from exemplification (Reder, Charney and Morgan, in press).

### CHANGING THE CURRENT DIRECTORY - CHDIR

The CHDIR command allows you to designate a directory as the "current" directory for a drive, so that the computer will automatically look there for files or sub-directories mentioned in your commands. You can designate a current directory for each disk drive independently.

#### FORMAT

```
CHDIR [loc and name of new current directory]
```

You can use the abbreviation CD in the command instead of typing CHDIR. [Location of new current directory] refers to the path to the directory you want to designate as the new current directory. The last directory name on the list should be the name of the directory you want to designate.

For example, the command below designates a subdirectory called PASCAL as the new current directory in drive B:

```
A) CHDIR B:\PROGRAMS\PASCAL (ENTER)
```

The first symbol in the path is a backslash (\). This means that the path to the new current directory starts with the root directory of the diskette in drive B. The path indicates that the root directory contains a subdirectory called PROGRAMS, and that PROGRAMS contains PASCAL, the directory you want to designate as the "new" current directory. As usual, the amount of location information you need to provide depends on which directory was last designated as the current directory for the drive.

To change the current directory back to the root directory, give a command like the following:

```
A) CHDIR B:\ (ENTER)
```

The backslash (\) in the commands above symbolize the root directory. So the command above changes the current directory for drive B to the root directory.

If you forget which directory is the current directory, the computer can remind you. Enter a CHDIR command without specifying a location. The computer will display the path from the root directory to the current directory or a backslash if you are still in the root directory.

Figure 1. Excerpt of manual illustrating RICH SYNTAX elaborations.

After they studied a version of the manual, subjects were asked to carry out a set of ordinary tasks on the computer, without referring back to the documentation. The subjects who had studied manuals containing syntactic elaborations worked significantly more quickly and issued significantly fewer commands. The conceptual elaborations did not significantly improve performance, perhaps because the selection of appropriate commands was fairly obvious for this particular set of tasks.

There is other evidence that examples strongly influence subjects' interpretation of procedural rules. LeFevre and Dixon (1984) and LeFevre (1985)