

# Technical NEWS

September 1997

FROM  
Dynamic Concepts Inc.

## No More LUST

As you know by now, dL4 does not have a Logical Unit Search Table (LUST) like UniBasic. This article discusses how to work in dL4 in the absence of LUST.

LUST under UniBasic introduced many inherent problems. For example, it was possible to kill a file, yet the file remained. It was also possible to build a file when that file already existed. Thus, it was difficult to address a file by name with any degree of certainty. The multiple directory searches needed by LUST cause substantial performance degradation. All of these problems are amplified in a network environment.

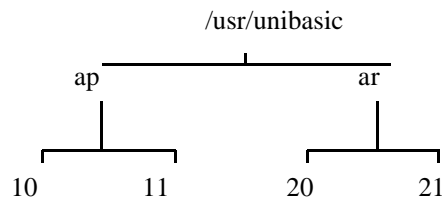
On a UNIX based system, directories or data files may be symbolically linked to construct a directory layout so that the existing files may be accessed via relative path names. The symbolic link allows support of multiple links to the same directory. On Windows 95/NT based systems, a directory layout should be constructed so that all data files may be accessed via relative path names. If the data files used by a Windows based system reside on a UNIX file server, then symbolic links can be used on a server.

LIBSTRING is a dL4 runtime parameter which contains a space separated list of directories to search when linking, CALLing, or CHAINing to program files. It is used by RUN, CALL, and CHAIN to find a program file. The LIBSTRING value is returned by the debugger's STATUS UNIT command and by the Msc\$(6) function.

An important distinction between UniBasic and dL4 is the way they find programs in their respective command line interpreter. Unlike UniBasic scope, which searches the LUST table to find a program, scope under dL4 does not search the LIBSTRING runtime parameter. Thus, a call to \$TRXCO using 10/programname will not work in dL4 unless

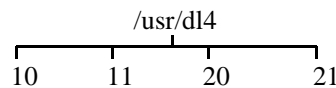
directory 10 exists relative to current working directory. This is because \$TRXCO simply passes the command to scope.

Let us take a closer look at a directory structure. In our hypothetical organization, all Accounts Payable (A/P) programs and data files reside in /usr/unibasic/ap/10 and /usr/unibasic/ap/11 directories, respectively. Furthermore, all Accounts Receivable (A/R) programs and data files reside in /usr/unibasic/ar/20 and /usr/unibasic/ar/21 directories, respectively. A diagram of this setup follows:



Let us also assume that all data files are opened as 11/datafilenames and 21/datafilenames. Similarly, all programs are CALLED or CHAINED as 10/programfile and 20/programfile. Hence, the LUST environment variable under UniBasic was set to /usr/unibasic/ap:/usr/unibasic/ar.

The same hypothetical organization's setup for dL4 may be represented as follows without the ap and ar directories:



The above directory structure can use symbolic links in a UNIX system, which allows both setups to coexist. Since Windows 95/NT does not support symbolic links, all files must reside in their proper directory. In both cases /usr/dL4 must be the working directory.

If you have any questions, or if your particular setup was not addressed here, please give us a call. We'll be happy to help!

**Check  
it  
Out!**

**dL4 2.3.1 for  
Windows 95/NT  
is now available  
from the Down-  
load Center on  
our Web site!  
This new ver-  
sion of dL4 cor-  
rects random  
"Driver Re-  
source Ex-  
hausted" error  
while opening a  
pipe.**

Telephone:  
800.822.2742  
714.448.8600



Email:  
support@dynamic.com

Technical  
**NEWS**

S e p t e m b e r 1 9 9 7

FROM  
**Dynamic Concepts Inc.**

**Telephone:**

800.822.2742

714.448.8600



**Email:**

support@dynamic.com