Notes on Converting to ACL 6*
SNeRG Technical Note 30

Stuart C. Shapiro
Department of Computer Science and Engineering
and Center for Cognitive Science
201 Bell Hall
University at Buffalo, The State University of New York
Buffalo, NY 14260-2000
shapiro@cse.buffalo.edu

May 8, 2001
Revised June 7, 2002

The intention is to convert Common Lisp source files so that they will run correctly either under Allegro CL version 6 or under previous versions of ACL or under other versions of ANSI Common Lisp. I will refer to the non-ACL6 versions as Old Common Lisp, or OCL.

The main issue is that under OCL, predefined Lisp symbols used uppercase letters for their names, and the OCL readers automatically uppercased all non-escaped characters. ACL6, however, uses lowercase letters for the names of predefined symbols, and leaves non-escaped characters in the case in which they appear in the files. Note that all versions of Lisp leave characters within strings in the original case. Therefore, the basic approach is to use symbols instead of names, and to put symbols in lower- or mixed- case. That way, when ACL6 leaves the characters as is, that will be correct, and when OCL uppercases characters, that will be correct also.

Packages Use of keyword symbols, such as `(find-package :user)` is fine. Use lower-case symbols instead of strings for packages or nicknames. When exporting symbols from an `:export` form within the `defpackage` form, use symbols in the keyword package, such as `(:export :foo)`, or, even better, use uninterned symbols, such as `(:export #:foo)`.

Pathnames Leave path strings, including extensions, as is. But I don't understand the paths in load-sneps that have uppercase extensions.

*These notes were written for people converting SNePS source-code files, but others might find them helpful also.

Other Issues In `#+`*feature* and `#-`*feature* the *feature* should be in lowercase. Note a *feature* is true, if it is a member of the list `*features*`.

The line

```
#+allegro (excl:set-case-mode :case-insensitive-upper)
```

must be deleted or commented out to preserve ACL6's case preservation. Change

```
#+allegro (proclaim '(notinline LAST))
```

to

```
#+allegro (proclaim '(notinline last))
```

since `last` is the name of a predefined function.

Leave global variables, such as `*NODES*` as is, but check carefully for consistency in the source code. If a function is defined with upper case or mixed case, leave it as is, again checking for consistency.

Very important: Change all occurrences of `T` to `t` and of `NIL` to `nil`.

The function `(intern string package)` insists on a string as its first argument. So instead of a form like `(intern "ATNIN" 'snepslog)`, use `(intern (symbol-name :atnin) :snepslog)`.

Compiler warnings are very helpful in finding uses of predefined function names in the wrong case, and inconsistent use of case in variables.

See also Franz's document section, "Allegro CL Case Modes," at `http://www.franz.com/support/documentation/6.0/doc/case.htm#introduction-1`