XCL-BRIDGE

By: Jan Pedersen (pedersen.PA @ Xerox.com)

XCL-BRIDGE is a module that assists in the transformation of ascii Common Lisp source files to Lisp managed files and vice versa. In the text-to-managed-file direction, user interaction is employed to repair read-in forms before establishing a resident image of a Lisp managed file. In the managed-to-text-file direction, a few simple transforms are employed to translate common filepackagecoms to equivalent Common Lisp forms.

All entry points are external to the "XCL" package.

(XCL:TEXT-TO-MANAGED-FILE *pathname filename* &key *(package* "USER"*)*
(readtable "XCL"*) (read-base* 10*) (combine-comments* t*)*) [Function]

Reads an ascii lisp source file named by "pathname" and converts it to a managed file with rootname "filename". The package, readtable, and read-base employed to read the ascii file may be specified via keywords arguments, or defaulted, as shown. If the reader environment arguments are defaulted and the source file has a emacs-style "mode line", then the package and read-base will be as indicated by the "mode line". The "combine-comments" keyword controls whether adjacent comments at the same ";" level should be combined when generating sedit-style comments for the converted file.

Note that forms are only read from the ascii lisp source file, not evaluated. It is assumed that the converted file should be made (via "il:makefile") and compiled before any evaluation should be attempted.

Text-to-managed-file proceeds incrementally and interactively to convert the specified file. First all the forms are read, and presented to the user for editing (via Sedit). If the user accepts this primary phase, a filecoms is generated and again, presented to the user for editing. If the user accepts the generated filecoms, a file (and its contained definitions) is instantiated, completing the conversion.

(XCL:MANAGED-TO-TEXT-FILE *filename pathname* &key *(package* "USER"*)*
(readtable "XCL"*) (print-base* 10*)*) [Function]

Prints a managed file, with rootname "filename", whose source definitions must be resident, to an ascii file "pathname" in a form suitable for reading by any Common Lisp reader. The read-print environment of the managed file may be overwritten via the keyword arguments "package", "readtable", and "print-base". Many Interlisp "filepackagecoms" are translated to their Common Lisp equivalents. For example, "il:declare\:" forms are transformed to "eval-when" forms and "il:files" forms are transformed to "require" forms. As an additional convenience, defdefiners are printed as equivalent defmacros.