

1 Issue 9 (Empty package local name)

1.1 Description

It is not clear whether it should be allowed to use "" as a local nickname, and in the case this is allowed, whether it should affect the :xxxx syntax.

1.2 Examples

```
(defpackage #:foo
  (:use #:cl)
  (:local-nicknames (" " #:cl)))

(in-package #:foo)

(package-name (symbol-package ':*package*))
; => "KEYWORD" (sbcl, ccl, ecl, abcl, clasp, lispworks)
; => "COMMON-LISP" (acl)

(package-name (symbol-package '||:*package*))
; => "KEYWORD" (ecl, clasp, lispworks)
; => "COMMON-LISP" (sbcl, ccl, acl)
; abcl errors
```

1.3 Current behavior

sbcl, ccl: :xxxx is read as a keyword; ||:xxxx is read as a symbol in the package named or nicknamed "".

ecl, clasp, lispworks: ||:xxxx and :xxxx are read as a keyword.

acl: :xxxx and ||:xxxx are read as a symbol in the package named or nicknamed "". "" is by default a global nickname for the #:KEYWORD package.

abcl: :xxxx is read as a keyword; ||:xxxx syntax cannot be read (attempts result in an error).

1.4 Proposal ALLOW-BUT-KEEP-KEYWORDS

The "" local nickname should be explicitly allowed. :xxxx should be always read as a keyword regardless of package names or nicknames. ||:xxxx should be read as a symbol in the package named or nicknamed by "".

1.5 Proposal ALLOW-FUN

The "" local nickname should be explicitly allowed. Both :xxxx and ||:xxxx should be read as a symbol in the package named or nicknamed by "".

1.6 Links

See WSCL issue 63.