1 Issue 3 (Local nicknames effect on DEFPACKAGE, MAKE-PACKAGE and others)

1.1 Description

It is not clear whether *local nicknames* of the *current package* should affect the resolution of package designators provided in `make-package` and `defpackage`, as well as other functions and macros taking a package designator as an argument (`in-package`, `add-package-local-nickname` and others).

1.2 Examples

```
(defpackage #:foo-a (:use) (:export #:x))
(defpackage #:foo-b (:use) (:export #:x))

(defpackage #:bar
  (:use #:cl)
  (:local-nicknames (#:foo-a #:foo-b)
                    (#:foo-b #:foo-a)))

(in-package #:bar)

(defpackage #:quux-1
  (:use #:foo-a))
(package-name (symbol-package 'quux-1::x))
; => "FOO-B"  (sbcl, ccl, acl, abcl, lispworks)
; => "FOO-A"  (ecl, clasp)

(make-package '#:quux-2 :use '(#:foo-a))
(package-name (symbol-package 'quux-2::x))
; => "FOO-B"  (sbcl, ccl, ecl, acl, abcl, clasp, lispworks)

(defpackage #:quux-3
  (:use)
  (:local-nicknames (#:foo #:foo-a)))
(let ((*package* (find-package '#:quux-3)))
  (package-name (find-package '#:foo)))
; => "FOO-B"  (ccl, ecl, acl, abcl)
; => "FOO-A"  (sbcl, clasp, lispworks)

(import (car (find-all-symbols (string '#:add-package-local-nickname))))
(defpackage #:quux-4
  (:use))
(add-package-local-nickname '#:foo '#:foo-a '#:quux-4)
(let ((*package* (find-package '#:quux-4)))
  (package-name (find-package '#:foo)))
; => "FOO-B"  (ccl, ecl, clasp, abcl, lispworks)
; => "FOO-A"  (sbcl)
```

```
(use-package '#:foo-a '#:quux-4)
(package-name (symbol-package 'quux-4::x))
; => "FOO-B"  (sbcl, ccl, ecl, clasp, abcl, lispworks)
```

1.3   Current behavior

sbcl, lispworks: only `:local-nicknames` clause is **not** affected by *local nicknames*.

ccl, acl, abcl: all options are affected.

ecl: only the `:local-nicknames` clause and keyword arguments (`:use` and `:local-nicknames`) are affected.

clasp: only keyword argument `:use` is affected.

Known exceptions in other functions/macros: sbcl: `add-package-local-nickname` is not affected. lispworks: `in-package` is not affected.

1.4   Proposal ALL-AFFECTED

All `defpackage` clauses (`:use`, `:local-nicknames`, `:import-from`, `:shadowing-import-from`) as well as all keyword arguments to `make-package` (`:use` and `:local-nicknames`) must be affected by the *local nicknames* of the *current package*.

All functions and macros taking a package designator as an argument must be affected as well.

A non-exhaustive list of affected functions and macros: `export`, `find-symbol`, `import`, `rename-package`, `shadow`, `shadowing-import`, `delete-package`, `with-package-iterator`, `unexport`, `unintern`, `in-package`, `unuse-package`, `use-package`, `do-symbols`, `do-external-symbols`, `do-all-symbols`, `intern`, `package-name`, `package-nicknames`, `package-shadowing-symbols`, `package-use-list`, `package-used-by-list`, `add-package-local-nickname`, `remove-package-local-nickname`, `package-local-nicknames`, `package-locally`