

matcher v1.7 – new features

brief

This document outlines the new features that have been introduced into the new matcher package (version 1.7).

match-> **match>>**

These differ in the way they deal with unbound match variables, `match->` is probably more often useful & we recommend using this by default

```
cg-user(80): (mlet ('(?a ?b) '(aa bb))
              (match-> '(?a ?b ?c)))
(aa bb ?c)
cg-user(81): (mlet ('(?a ?b) '(aa bb))
              (match>> '(?a ?b ?c)))
(aa bb nil)
```

#?

In addition to accessing variables from the matcher name-space `#?` works as an abbreviation to access patterns from lists. In this respect it functions in a similar way to `match->` [v. 1.7f] (or like `match>>` upto v 1.7e).

```
(mlet ('(?a ?b) '(aa bb))
      #?(?a ?b ?c))
➔ (aa bb ?c)

(setf xxx '(?a ?b blah))
➔ (?a ?b blah)

#?xxx
➔ (?a ?b blah)
```

:not
:guard

The existing `forevery` and `all-present` forms may now include `:not` and `:guard`.

:not pattern – requires pattern not to be present for the match to succeed...

```
(defparameter db1
  '((isa b1 box)      (isa b2 box)      (isa b3 box)      (isa b4 box)
    (color b1 red)   (color b2 red)   (color b3 blue)   (color b4 blue)
    (size b1 large)  (size b2 small)  (size b3 small)  (size b4 small)
    (supports b1 b2) (supports b2 b3)
  ))

(forevery ('((isa ?b box) (isa ?a box) (:not (color ?b red))
           (:not (color ?b red)) (size ?a small))
          db1)
  (match>> '(?a ?b)))
→ ((b2 b3) (b3 b3) (b4 b3) (b2 b4) (b3 b4) (b4 b4))

(forevery ('((isa ?b box) (:not (color ?b red))) db1) #?*match)
→ (((isa b3 box)) ((isa b4 box)))

(forevery ('((isa ?b box) (color ?b red)
           (isa ?a box) (:not (size ?a small)))
          db1)
  (match>> '(?a ?b)))
→ ((b1 b1) (b1 b2))
```

:guard predicate – requires the predicate to be non-nil for the match to succeed...

```
(forevery ('((isa ?b box) (isa ?a box) (color ?b ?c)
           (size ?a small) (:guard (not (eq #?c 'red)))) db1)
  (match>> '(?a ?b)))
→ ((b2 b3) (b3 b3) (b4 b3) (b2 b4) (b3 b4) (b4 b4))

(forevery ('((isa ?b box) (color ?b ?c)
           (:guard (not (eq #?c 'red)))) db1) #?b)
→ (b3 b4)

(forevery ('((isa ?b box) (color ?b ?c) (:guard (eq #?c 'red))) db1) #?b)
→ (b1 b2)

(forevery ('((isa ?b box) (color ?b red) (size ?a ?s)
           (:guard (not (eq #?s 'small)))) (isa ?a box))
          db1)
  (match>> '(?a ?b)))
→ ((b1 b1) (b1 b2))
```