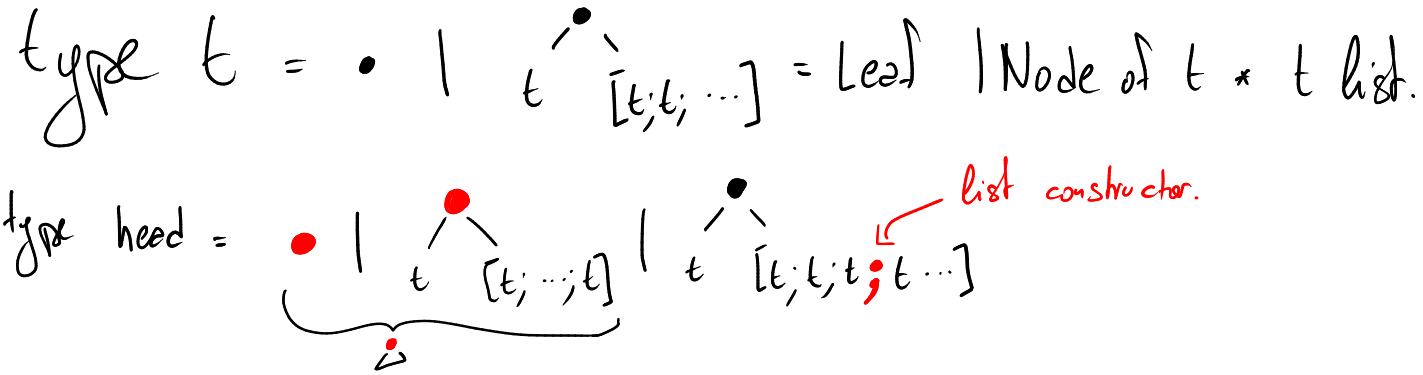
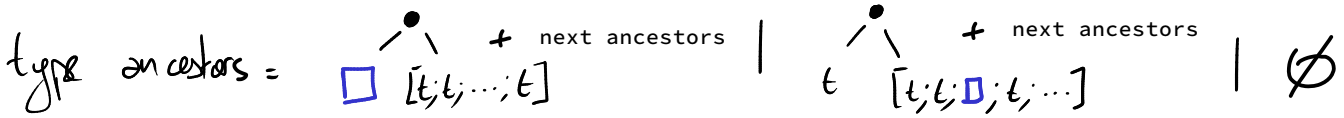


= Raw of t | InNode1 of t * t List.Zipper.t

(□ = hole)



= Node0 of hole * t list * ancestors | Node1 of t * t List.Zipper.d0 * ancestors | NoAncestor

type zipper = head * ancestors

---

(1) go-up , [] =

(2) go-up , [[t;t;…;t] ; . . .] = new pointer is here [t;t;…;t] , [. . .]

(go_up (Raw tree, Node0 (Hole, list_of_trees, ancestors)) = Up (Raw (Node (tree, list_of_trees)), ancestors))

(3) go-up , [t [t;t;□;…;t] ; . . .] = new pointer is here t [t;t;□;…;t] , [. . .]

(go_up (Raw tree, Node1 (tree', list_d0, ancestors)) = Up (InNode1 (tree', List.Zipper.plug_d0 tree list_d0), ancestors))

the hole is hidden in here

New function!

(4) go-up t [t;t;t;t…] , ancestors = the pointer moved up in the list t [t;t;t;t…] , ancestors

(5) go-up t [t;t;t;t…] , ancestors = the pointer is now on Node t [t;t;t;t…] , ancestors

```
( go_up (InNode1 (tree, list_zipper), ancestors) = match List.Zipper.go_up list_zipper with
                                                   | Up new_zipper -> Up (InNode1 (tree, new_zipper), ancestors)
                                                   | Top list -> Up (Raw (Node tree, list), ancestors)
```

(4)
(5)