[with (x 5) (adder (fun (x) (fun (y) (+ x y)))) (z 3)]

var-with var with body with

(interp (subst adder (fun (x) (fun (y) (+ x y))) (z 3))
id x
num 5)

subst (adder (fun (x) (fun (y) (+ x y)))) (z 3) (id x) (num 5))

interp '(fun (idx) (fun (idy) (add id x idy))) (idz num 3)

[fun (x) (fun (idy) (add idx idy)) (idz num 3)]

bound id bound body expr

[app (fun (x) (fun (y) (+ x y))) (z 3))

fun-expr arg expr

(local ([define fun val (interp (fun (x) (fun (y) (+ x y))))]))

* = (interp (fun (x) (fun (y) (+ x y))))

bound id bound body

(interp (fun (y) (+ x y)))

bound id bound body

i.e. {fun (x) (fun (y) (+ x y))}
es una expr

fun-val = (fun (x) (fun (y) (+ x y)))

(interp (subst (fun_body ((fun (x) (fun (y) (+ x y))))) → selecciona fun (y) (+ x y)
(fun-arg (fun (x) (fun (y) (+ x y)))) → selecciona x
(interp idz num 3)))) → num 3 = z

hasta aqui unicamente se subst de idx con num 5, y se acarrea (idz num 3), también se checa que adder este definida en FWAE

[with (y 10) (add 5 (adder x))]

var-with val with body with

(interp (subst (add 5 (adder x))
id y
num 10))

[interp (add5 (adder x))]

como ya se vio adder es de tipo FWAE entonces toma el $x \to 5$ y procede a ejecutar adder con idx = num 5 y arrastra (idy num 10)

los ids acarreados hasta aquí son $id_y = 10$ $id_x = 5$ $id_z = 3$

[with ((x (+ 10 z)) (y (add5 6)))]

(interp (subst (idy (add5 0)) ⇒ ①

id x
(interp (add num 10 id z)))

(((5)(f(0)(+ 5 0)

num 5

y=0 x=5

donde id z alcanza el valor acarreado num 3

(subst (idy (add5 0)) (idy) (add5 0))

expr sub-id val

donde (add 5 0) = 5

[id (y) (if (symbol? y (idy)) (num 0)) expr]} ⇒ num 5 = id y

sustituyendo en ①

acarreamos y = 5
x = 13

= (add num 10 num 3)
= (+ (num 10) (num 3))
= (+ 10 (num 3))
= (+ 10 3) = 13 = x

[interp (add (add idy idx) idz))]
= [add (add idy idx FWAE?) (idz FWAE?)] se pregunta si el lado izq y derecho son de tipo FWAE
= [add (l r) (+ interp (add idy idx) (interp idz))]

① = (+ interp (idy) (idx)) ⇒ ⑪ = (+ idy idx)

como ambos lados son de tipo FWAE sustituye las variables acarreadas

y = 5
x = 13
z = 3

= (+ (add num 5 num 13) (num 3))
= + (+ (num 5) (num 13)) (num 3))
= + (+ 5 (num 13)) (num 3))
= + 18 3 = 21