load "dmaps"

open DMap

define f := lambda (i) [(string->id ("s" joined-with (val->string i))) i]

define L := (from-to 1 5)

define sample-map := (make-map (map f L))

# So sample-map maps 's1 to 1, ..., 's5 to 5.

{eval sample-map at 's1}
{eval sample-map at 's2}
{eval sample-map at 's3}
{eval sample-map at 's4}
{eval sample-map at 's5}

# And this should give the default value 0:

{eval default sample-map}

let (ml := [77 [{'x --> 1} {'y --> 2}]]; m2 := [78 [{'y --> 2} {'x --> 1}]]
{eval (agree-on ml m2 ['x 'y])}

define sm1 := [0 [{'x --> 1} {'y --> 2} {'z --> 3}]]
define sm2 := [0 [{'y --> 2} {'z --> 3} {'x --> 1}]]

{eval sm1 |^ ['z 'y]}