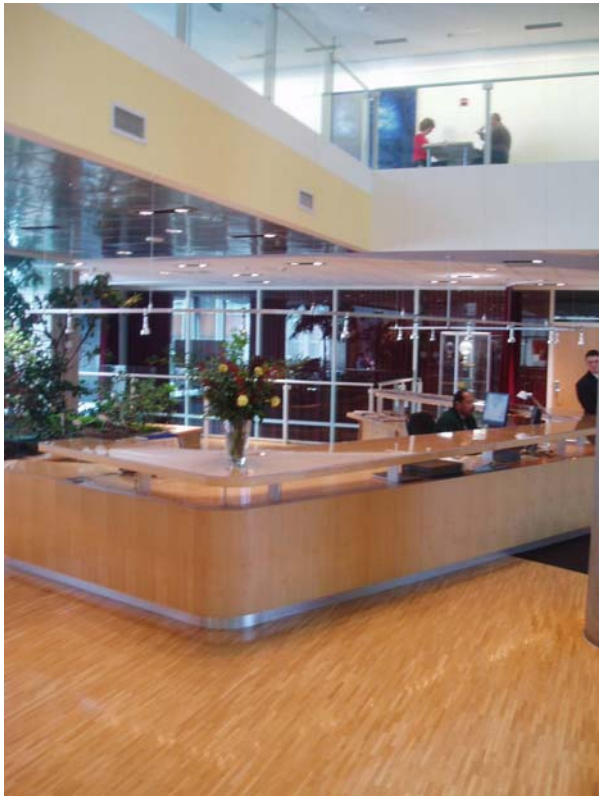
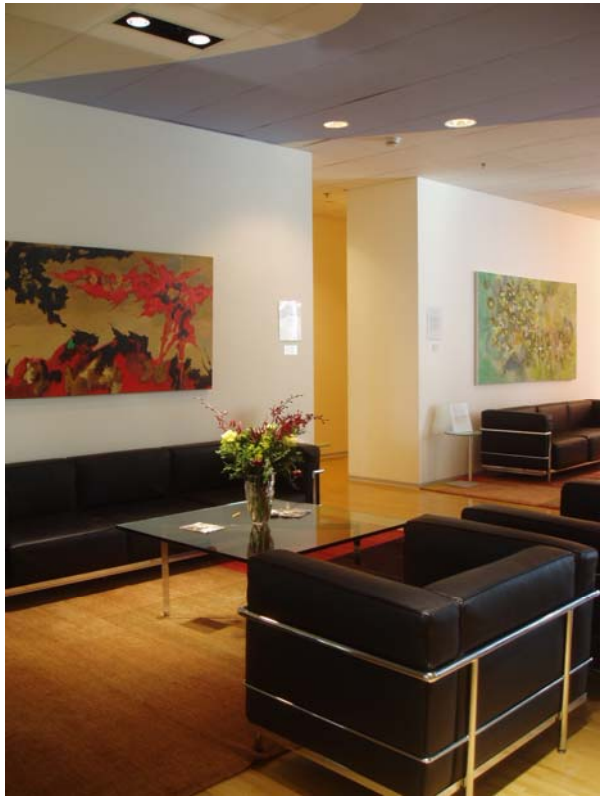
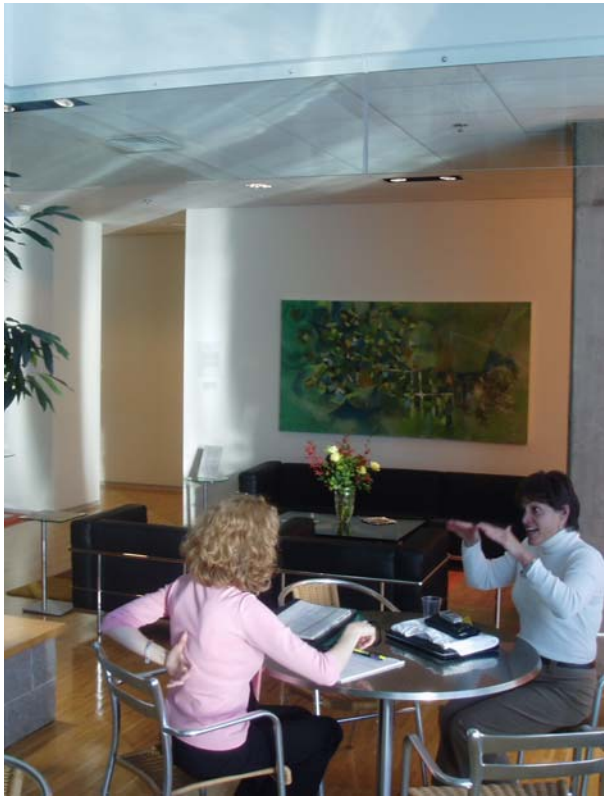
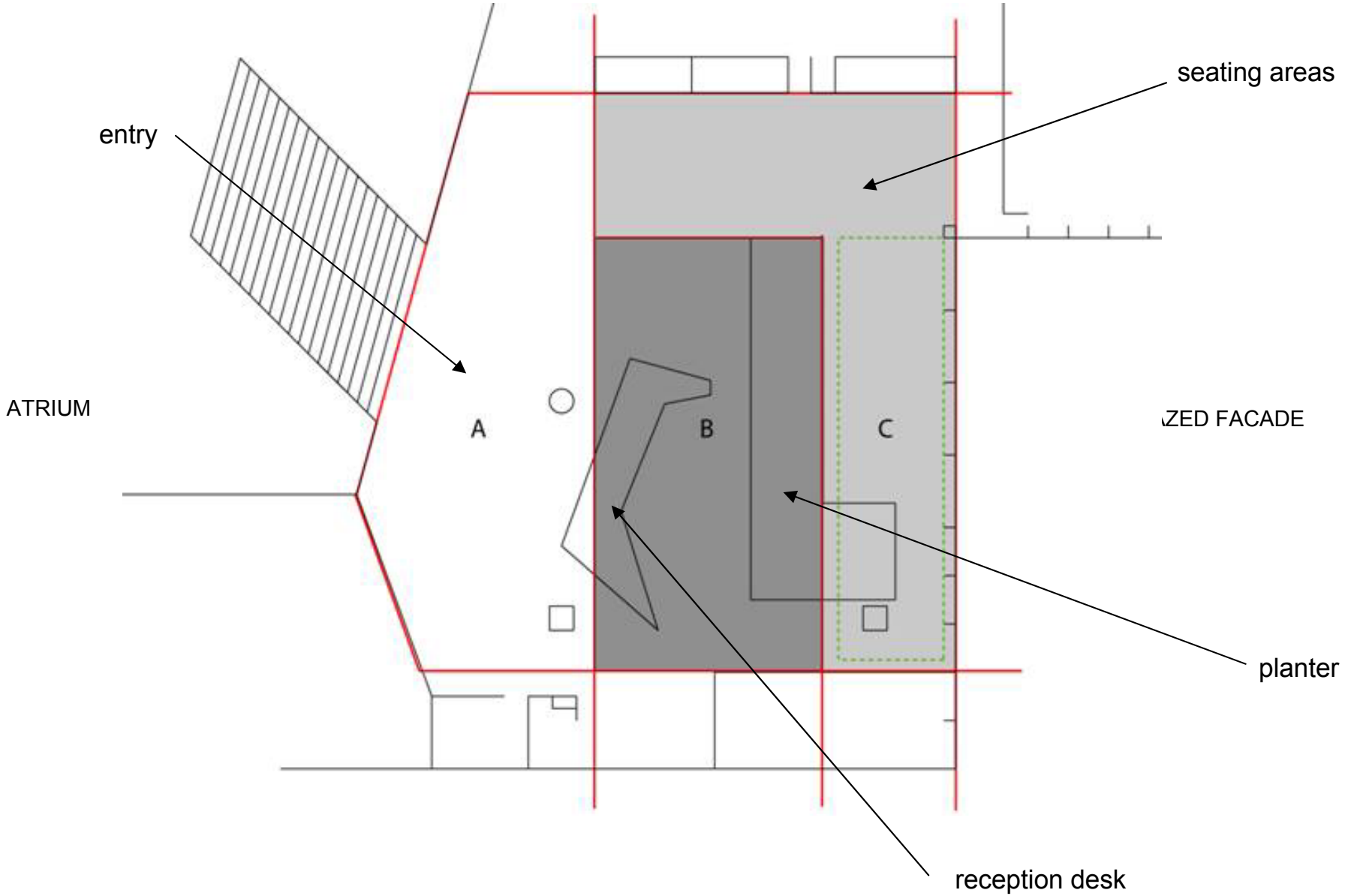
Daylighting design project
zone 2E: Genzyme Reception
Nicole Vlado & James Forren



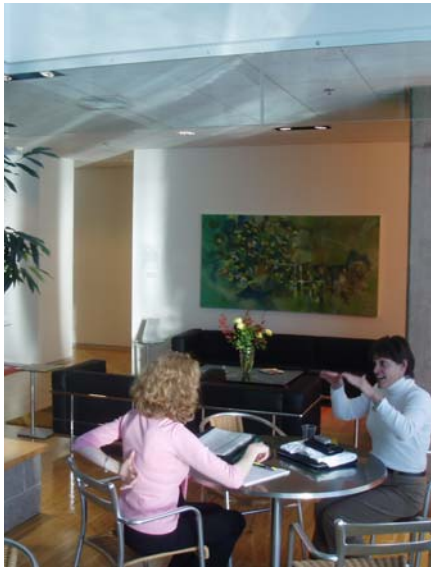




GENZYME RECEPTION

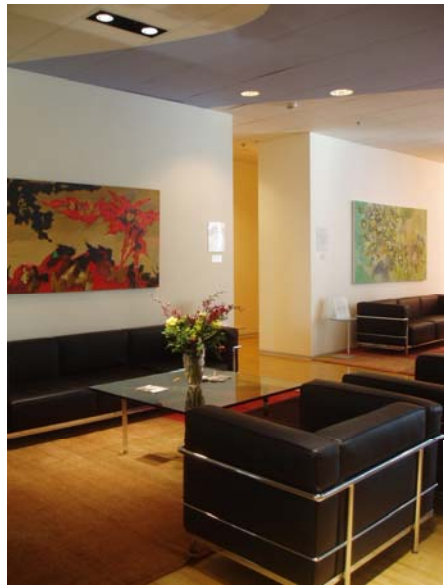


OBSERVATIONS

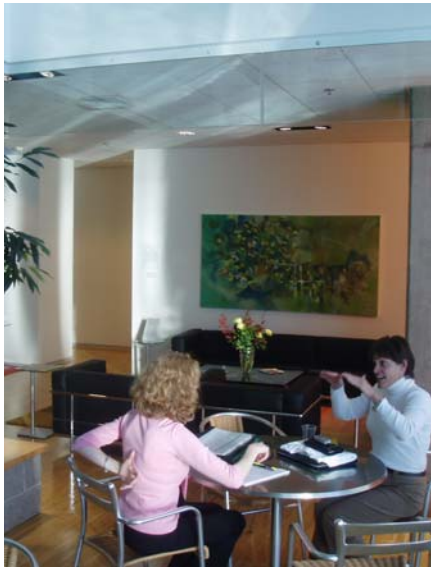






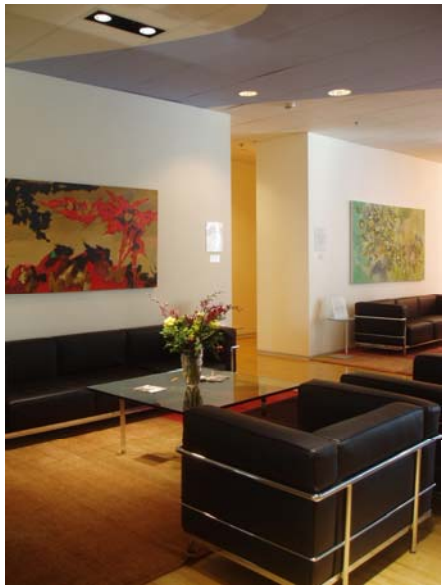
OBSERVATIONS



materials used:
polished wood floors, matte white walls (ρ=0.85)
brushed aluminum tables (near façade), black leather couches, glass tables
use of planter to protect entry and reception desk from direct light

spatial aspects:
double story height @ glazed façade
drop ceiling over desk and seating area (⊥ to façade)
receives ambient light from atrium (west)

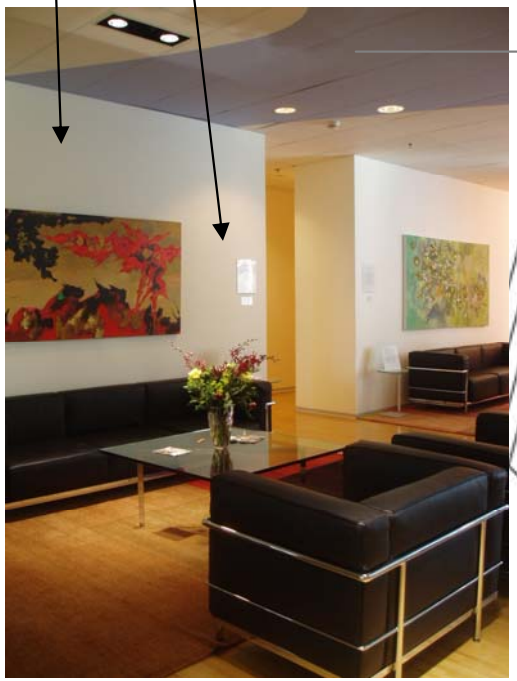
lighting observations:
use of reflective blinds (computer-operated)
spotlights recessed within drop ceiling

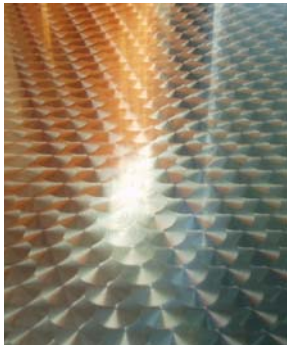


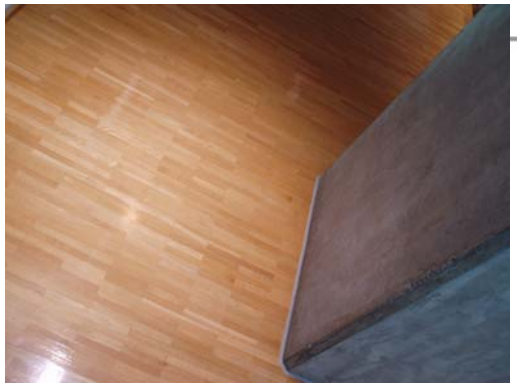
OBSERVATIONS
large eastern façade protected with reflective blinds

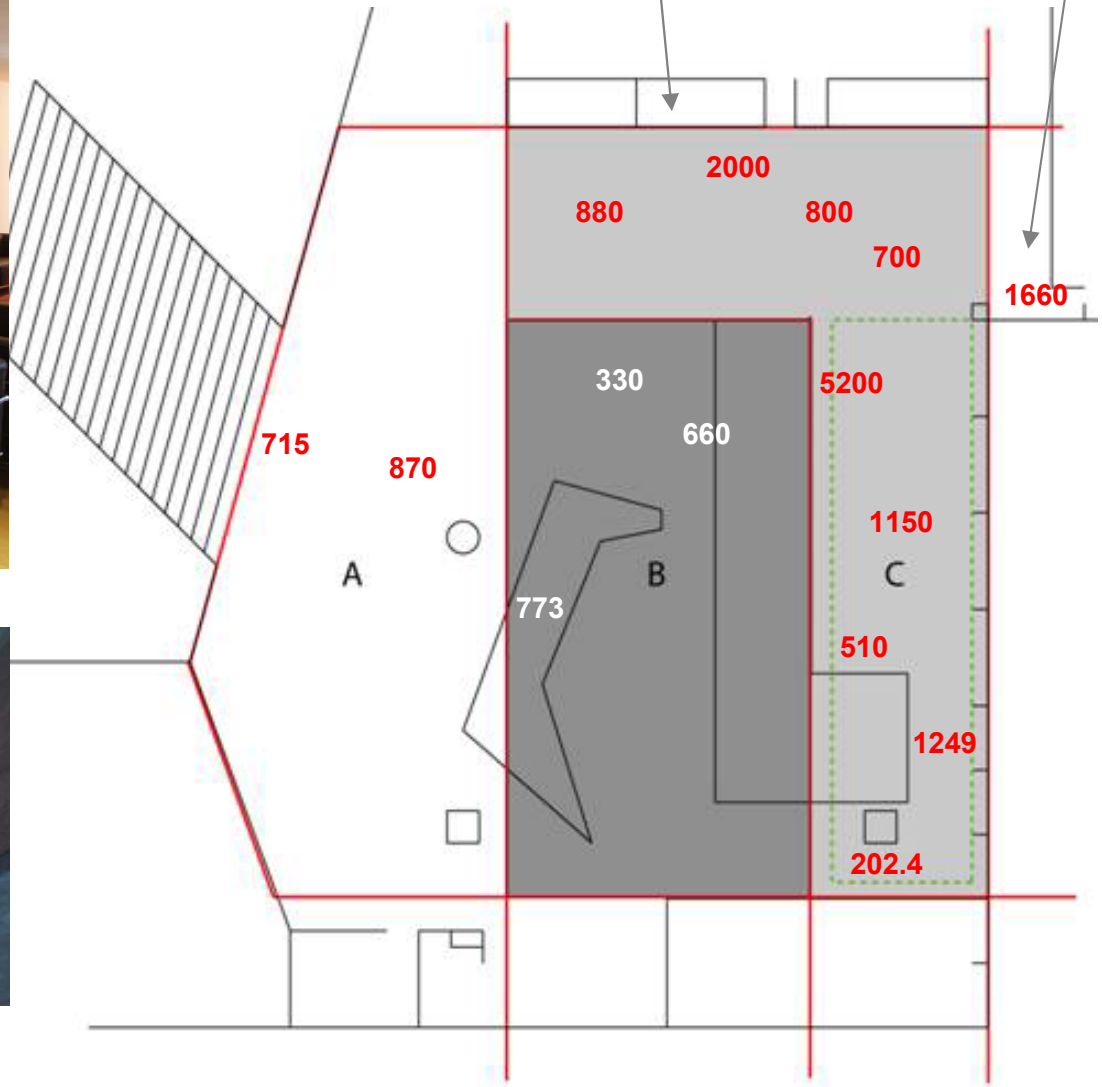
problems: reflective glare, no variation along length, bad view

OBSERVATIONS







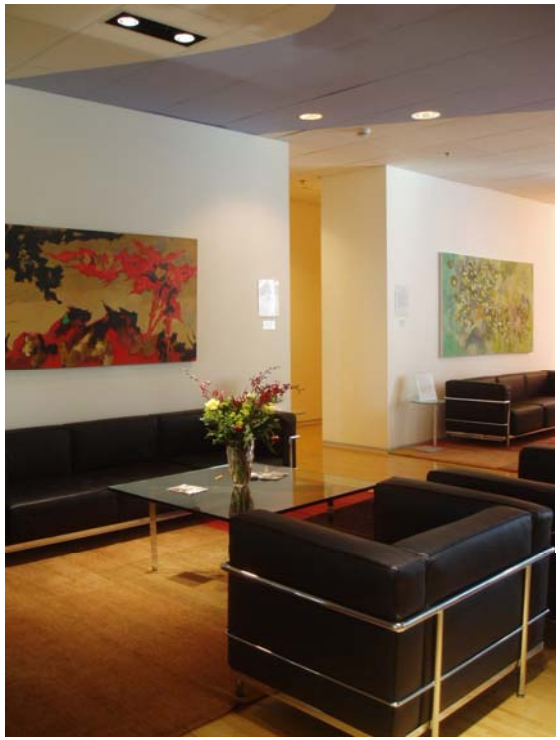


GOALS

to prevent reflective glare (currently produced by reflective blind system)
* to provide less direct/more diffuse light throughout space

to block out view to the outside while providing daylighting

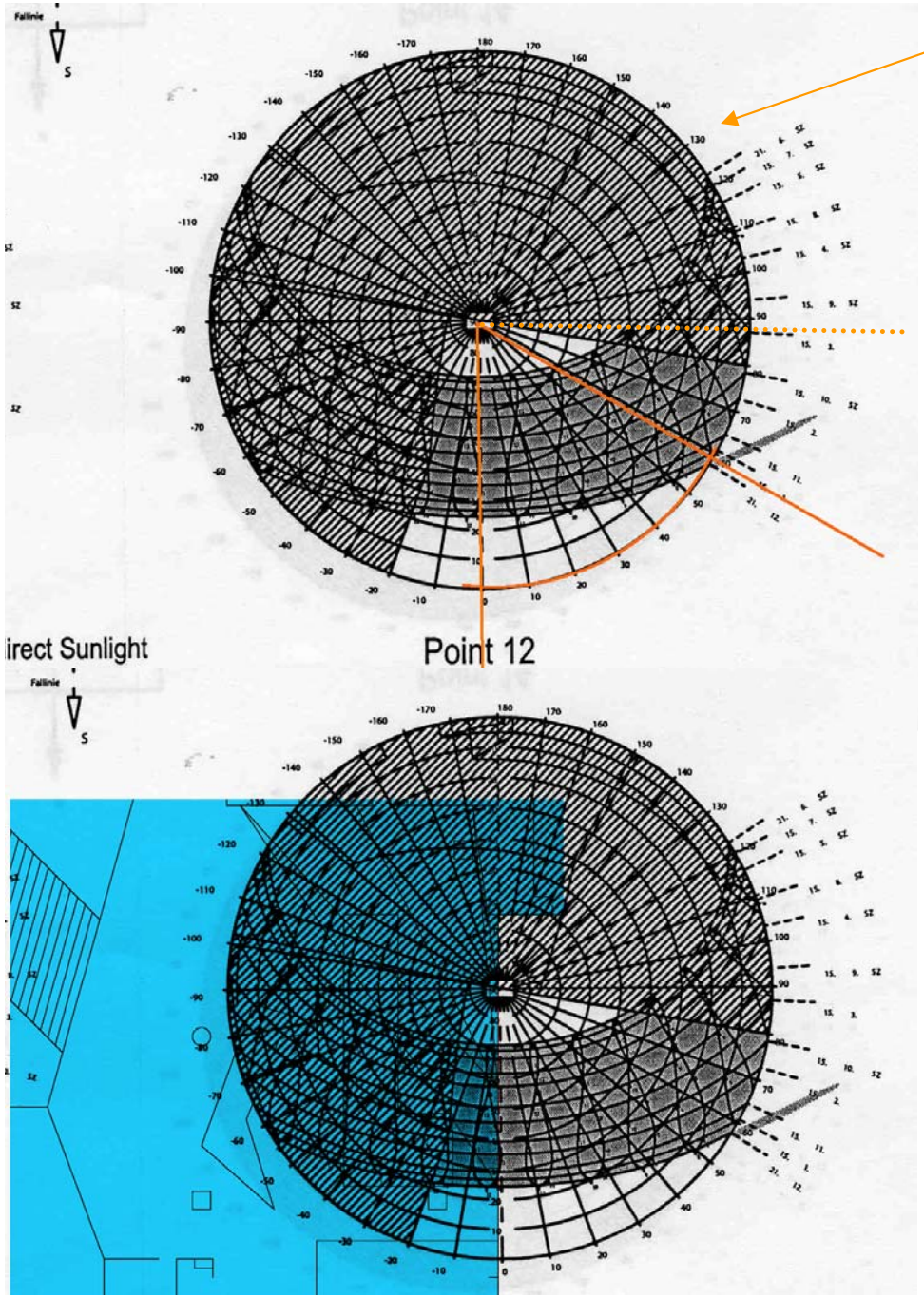
to optimize daylight received along interior wall ⊥ to glass façade
* to achieve varied illuminance throughout the day: 50 lux (MIN) – 500 lux (MAX)
* while maintaining comfortable condition at seating area directly in front of façade

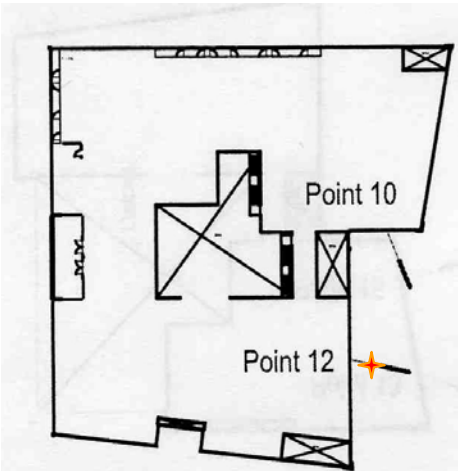


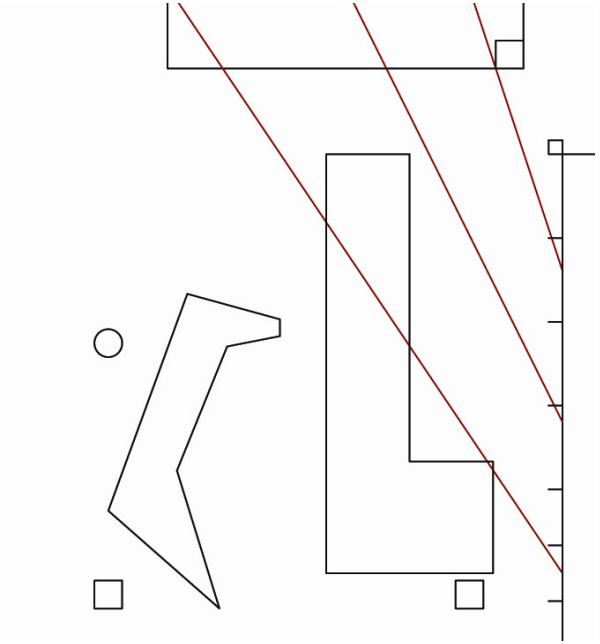
STRATEGIES

direct light from façade towards ⊥ interior wall

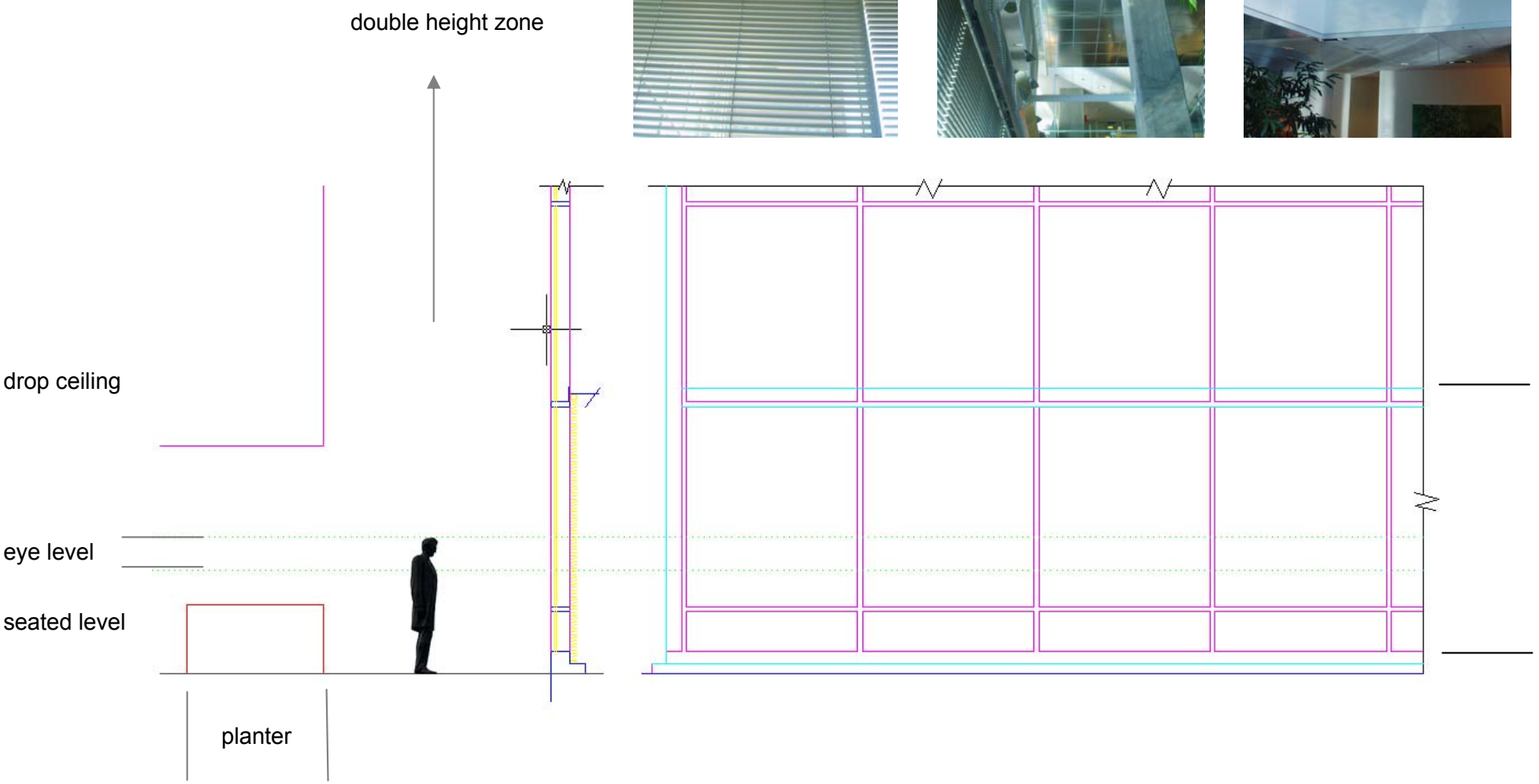
*light will enter the eastern façade such that it will be oriented towards the perpendicular wall during the late morning.



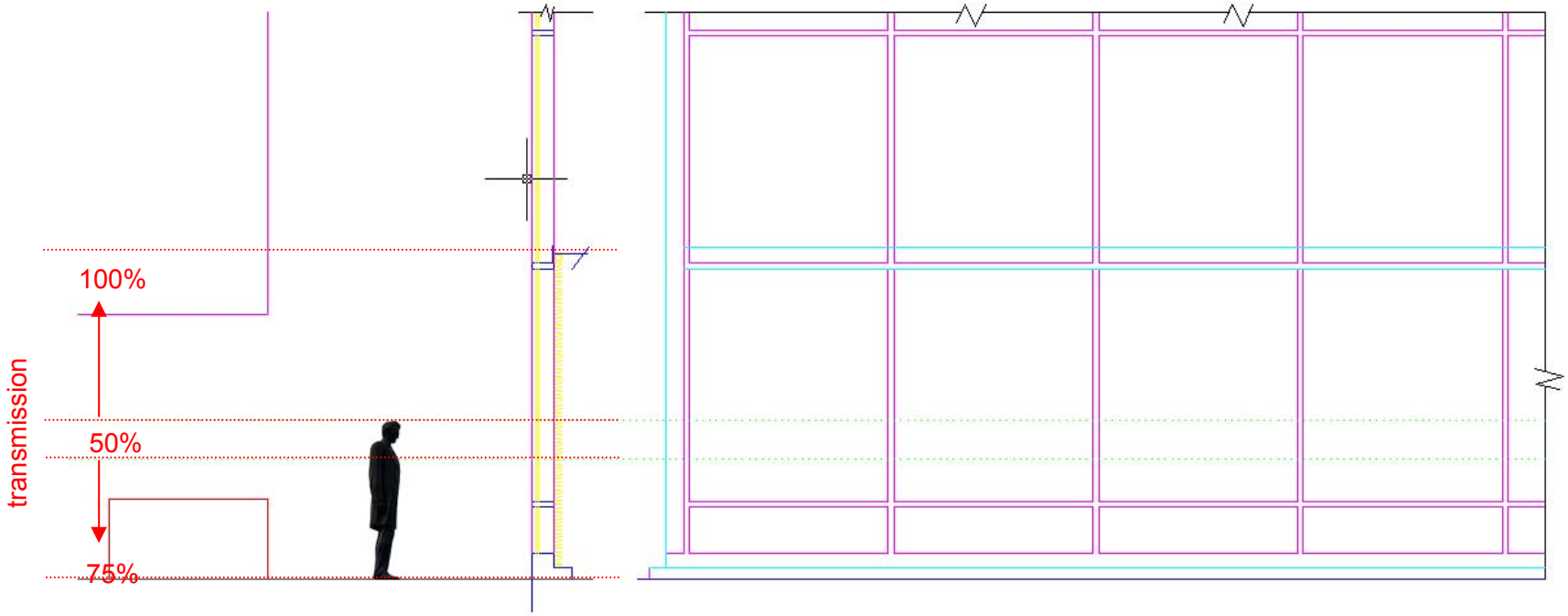




STRATEGIES
direct light from façade towards ⊥ interior wall



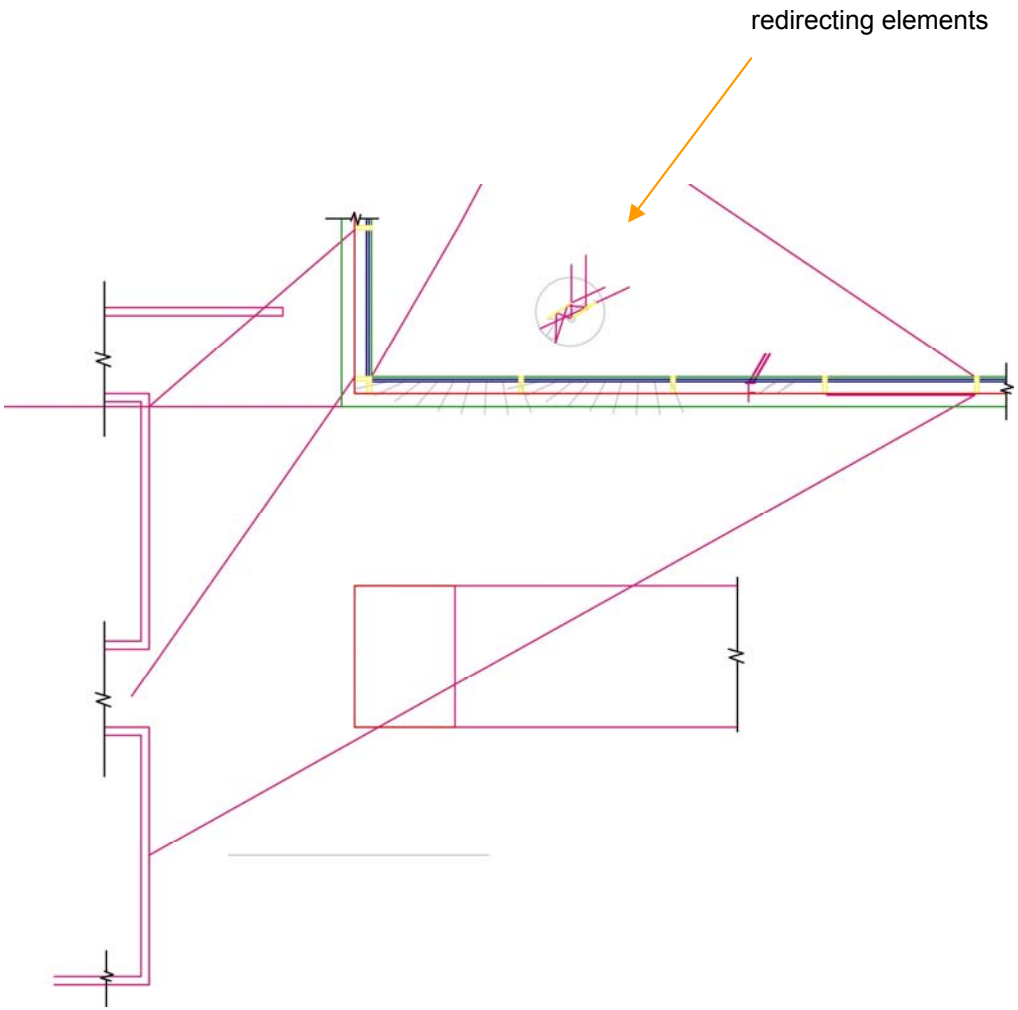
*step 1: remove existing reflective blind system to reduce glare
proposal: replace with a fixed system that diffuses and redirects daylight

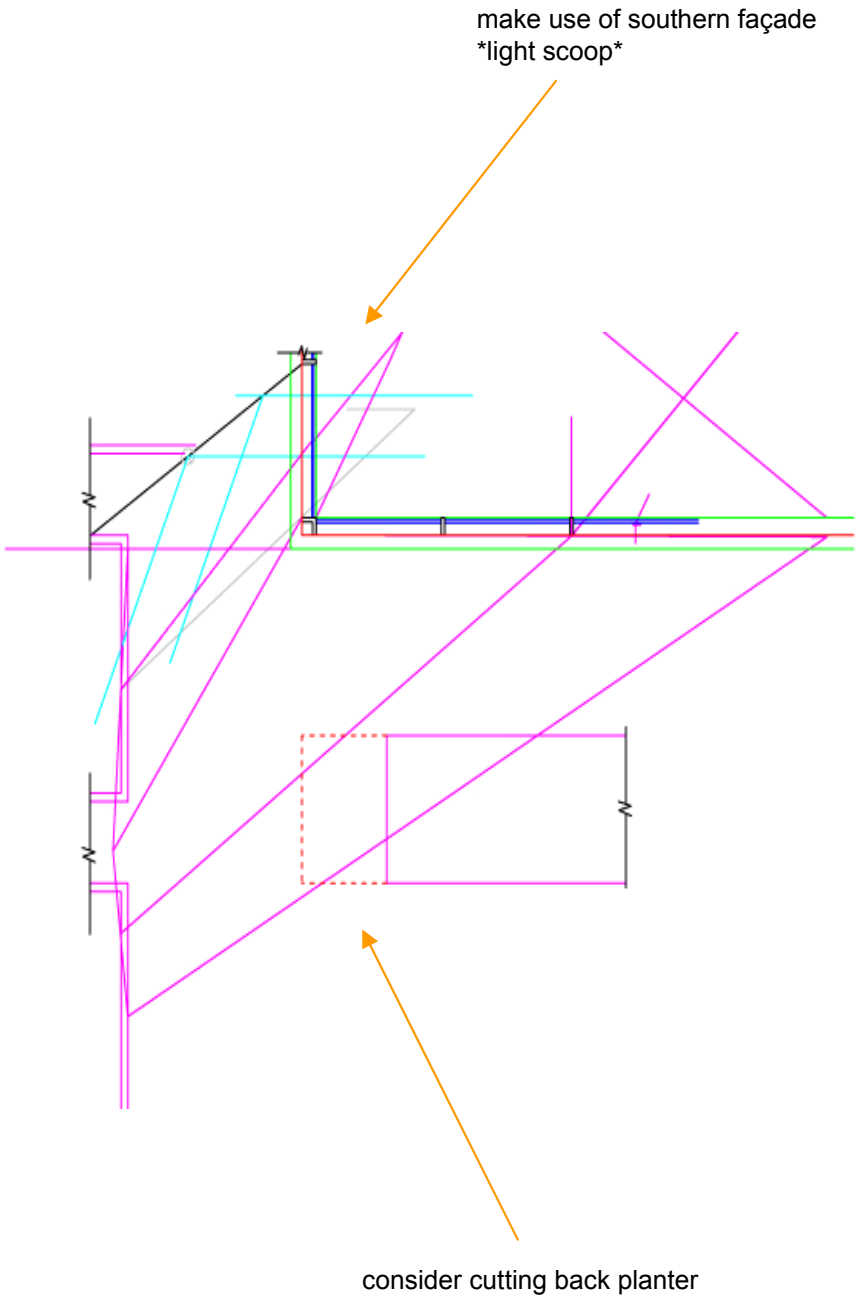


*transmission will be reduced to 68%

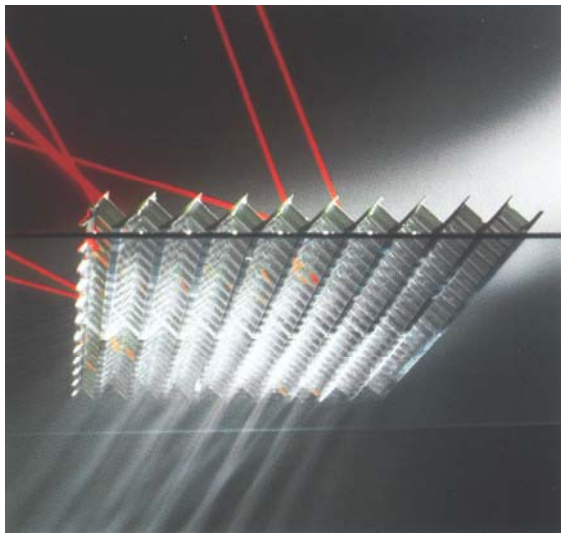
STRATEGIES

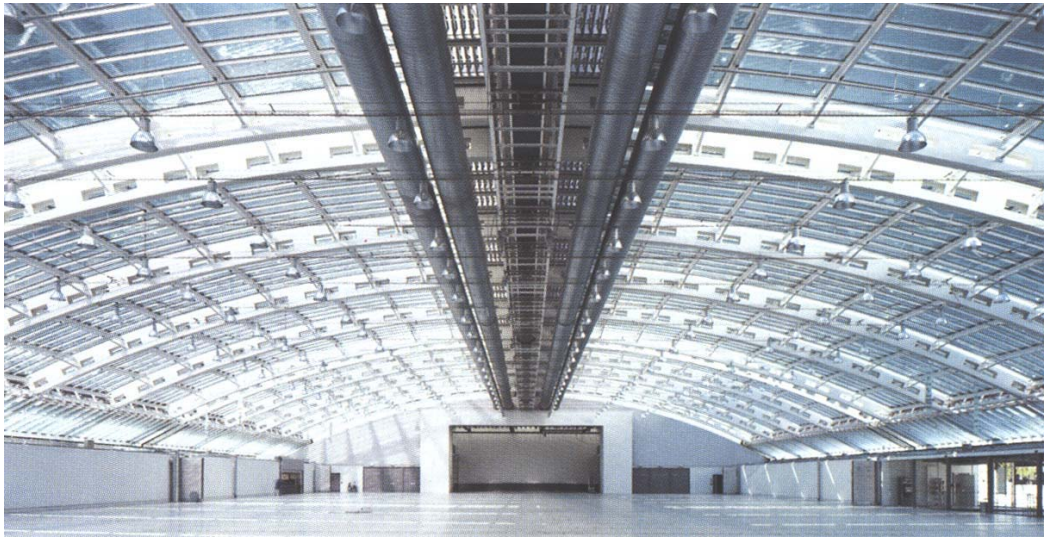
direct light from façade towards ⊥ interior wall



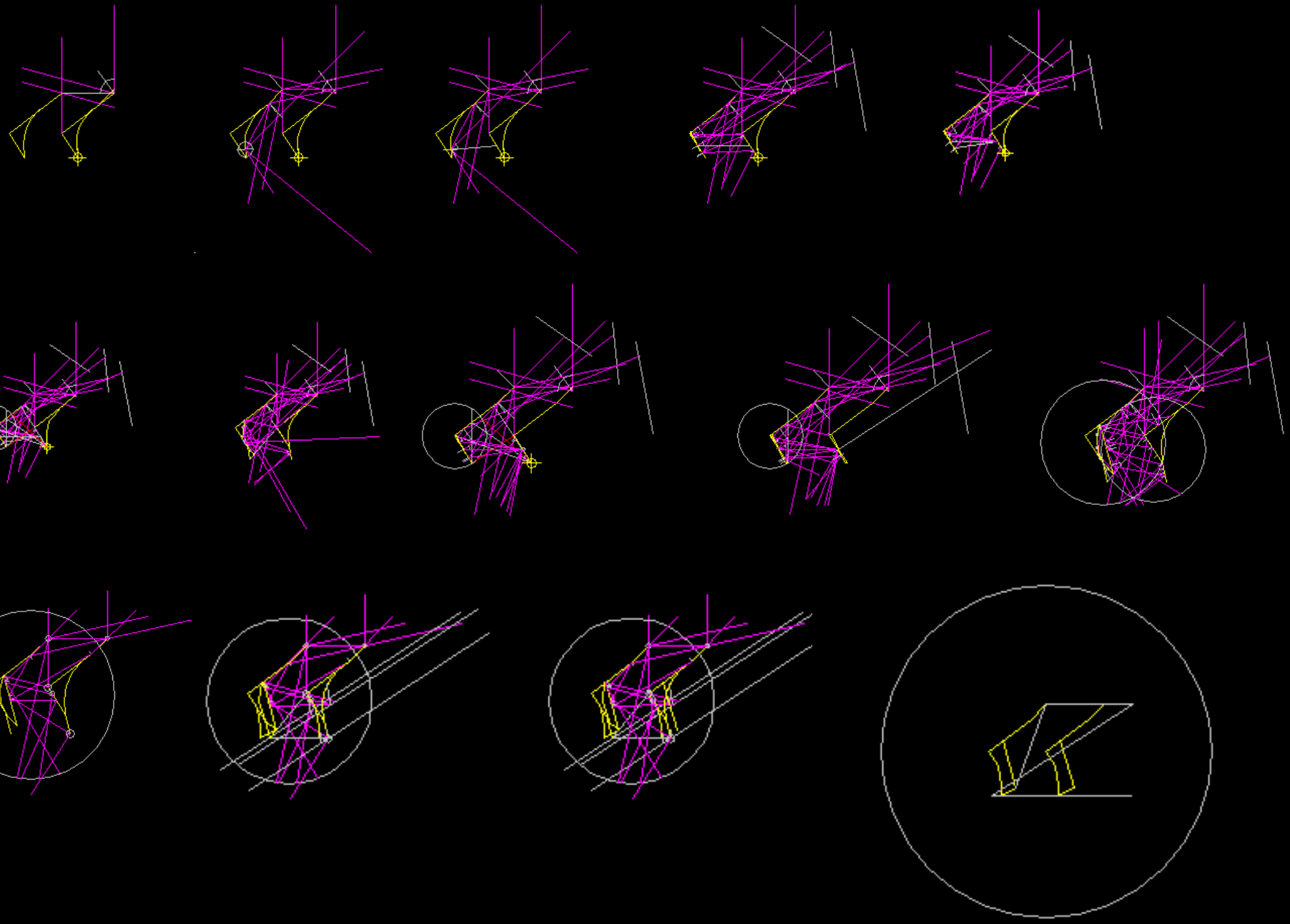


DESIGN PRECEDENTS
diffusing elements

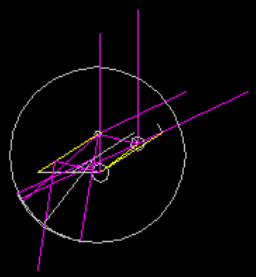


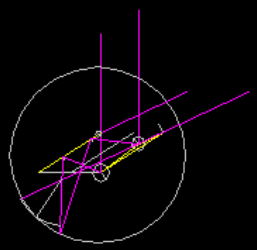


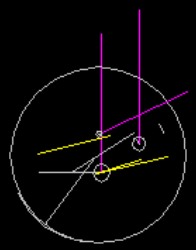
DESIGN STRATEGIES
diffusing elements

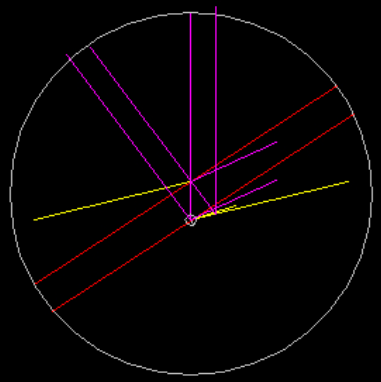


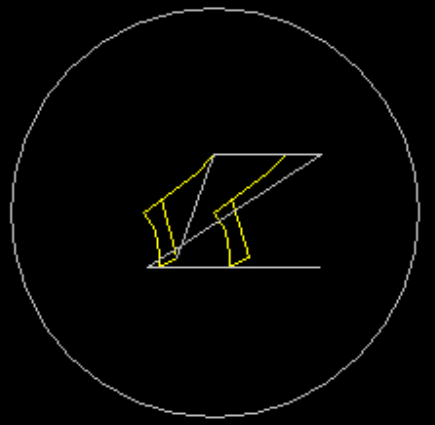
DESIGN STRATEGIES
diffusing elements

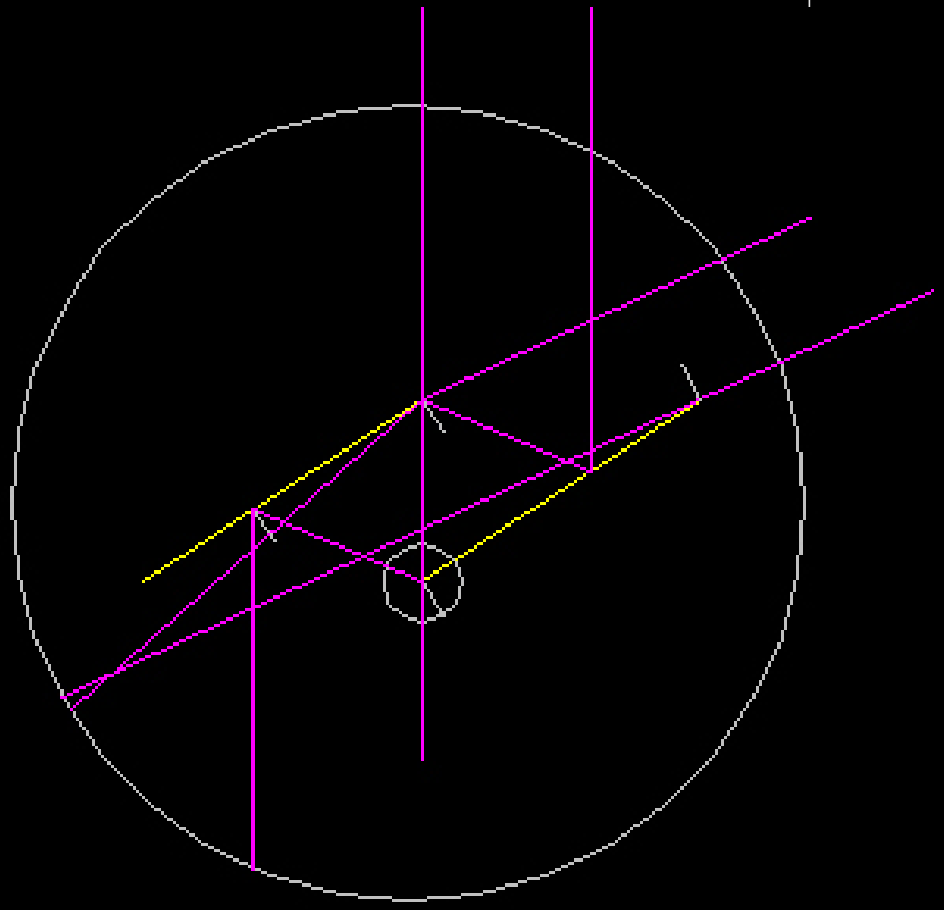




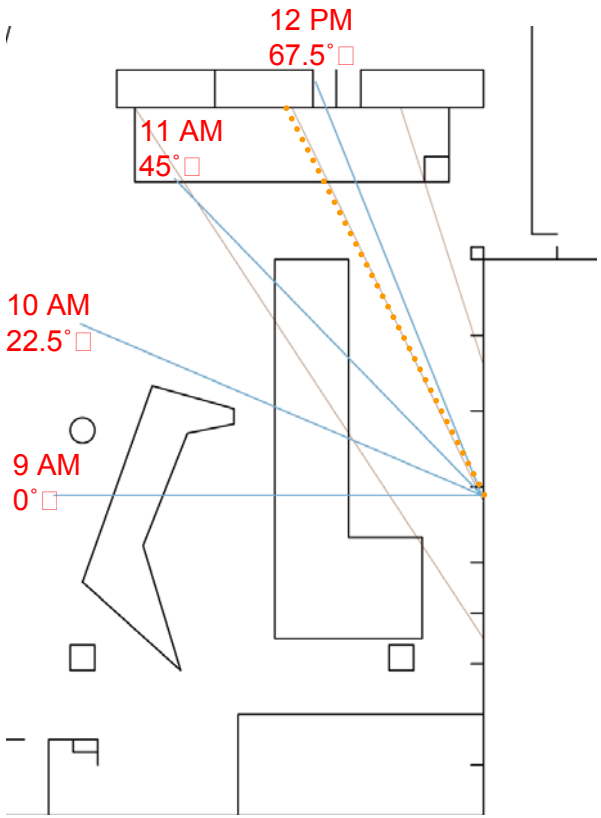








CALCULATIONS



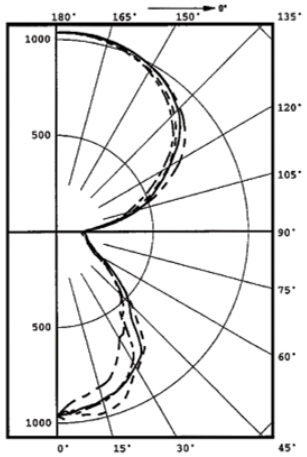
June	E1 (overcast)	E1 (clear)	refl E	E alpha	E recieved	E R clear
overcast	176.445347	1764.45347	132.33401	30.3348214	13.7416741	137.416741
	489.688214	4896.88214	367.266161	50.8928571	38.6785714	386.785714
	867.358611	8673.58611	650.518958	63.6830357	60.562567	605.62567
	1158.23205	11582.3205	868.674035	66.2946429	65.6316964	656.316964
			9:00 AM	68%	9.34433839	93.4433839
			10:00 AM		26.3014286	263.014286
			11:00 AM		41.1825455	411.825455
			12:00 PM		44.6295536	446.295536

December	E1 overcast	E1 clear	refl E	E alpha	E recieved	E clear
overcast	201.449315	2014.49315	171.231918	34.3794643	15.5738973	155.738973
	288.476685	2884.76685	245.205182	57.6785714	43.8357143	438.357143
	211.449557	2114.49557	179.732124	72.1741071	68.6375759	686.375759
	179.433789	1794.33789	152.518721	143.04133	141.610917	1416.10917
				68%	10.5902502	105.902502
					29.8082857	298.082857
					46.6735516	466.735516
					96.2954234	962.954234

*calculations assume NO protection on glazing
sky luminance: 300 cd/m² (overcast), 3000 cd/m² (clear)
our system assumes 68% transmission

DESIGN STRATEGIES
artificial lighting: operable table lamps

Candela Distribution Chart



CANDELA DISTRIBUTION

	0.0	90.0	180.0	270.0	FLUX
0	962	962	962	962	
5	936	947	961	893	88
15	975	908	920	841	256
25	893	856	847	735	392
35	777	762	634	604	446
45	551	527	463	476	394
55	347	345	313	299	291
65	208	213	193	189	203
75	162	167	159	152	172
85	147	149	143	132	159
90	141	145	156	147	
95	195	211	191	189	225
105	403	436	375	427	441
115	582	594	540	647	598
125	746	765	722	813	683
135	863	893	844	906	677
145	941	963	927	966	596
155	991	1006	985	1006	461
165	1022	1031	1018	1027	290
175	1036	1040	1034	1036	99
180	1037	1037	1037	1037	

DESCRIPTION: Berkeley

LAMP: 2 - 55 2D CFL

BALLAST: Electronic

POWER FACTOR: NPF

REFLECTOR: White

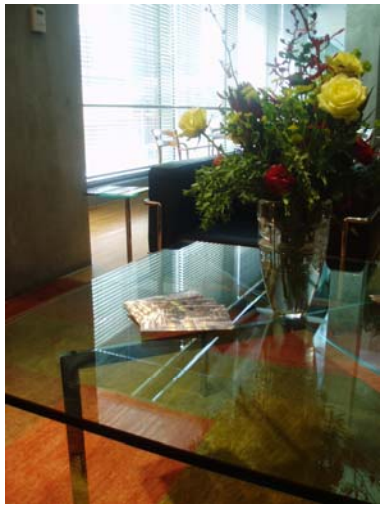
LENS: None

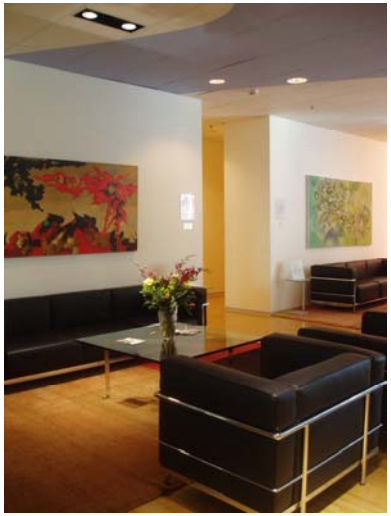
patent pending

ZONAL LUMEN SUMMARY

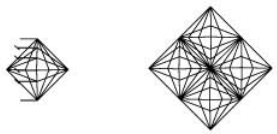
ZONE	LUMENS	%LAMP	%FIXT
0-30	736	9.4	11.4
0-40	1183	15.2	18.3
0-60	1867	23.9	28.8
0-90	2402	30.8	37.1
90-120	1264	16.2	19.5
90-130	1947	25.0	30.1
90-150	3220	41.3	49.8
90-180	4071	52.2	62.9
0-180	6473	83.3	100.0

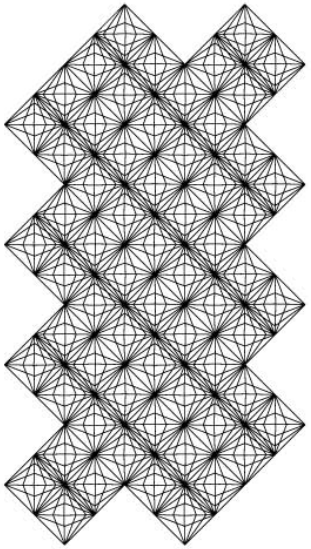
Total luminaire efficiency = 83.0%
CIE type = Semi - Indirect

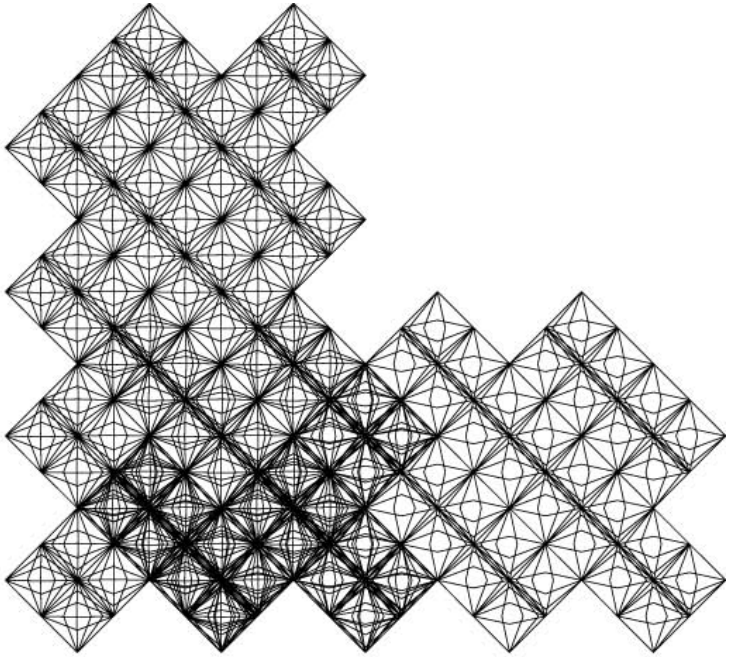


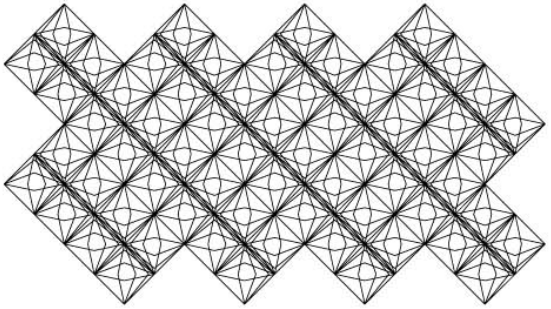


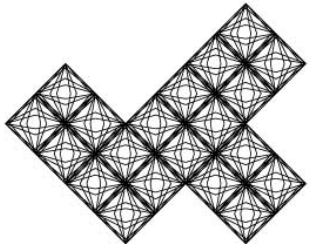
DESIGN STRATEGIES
cell division



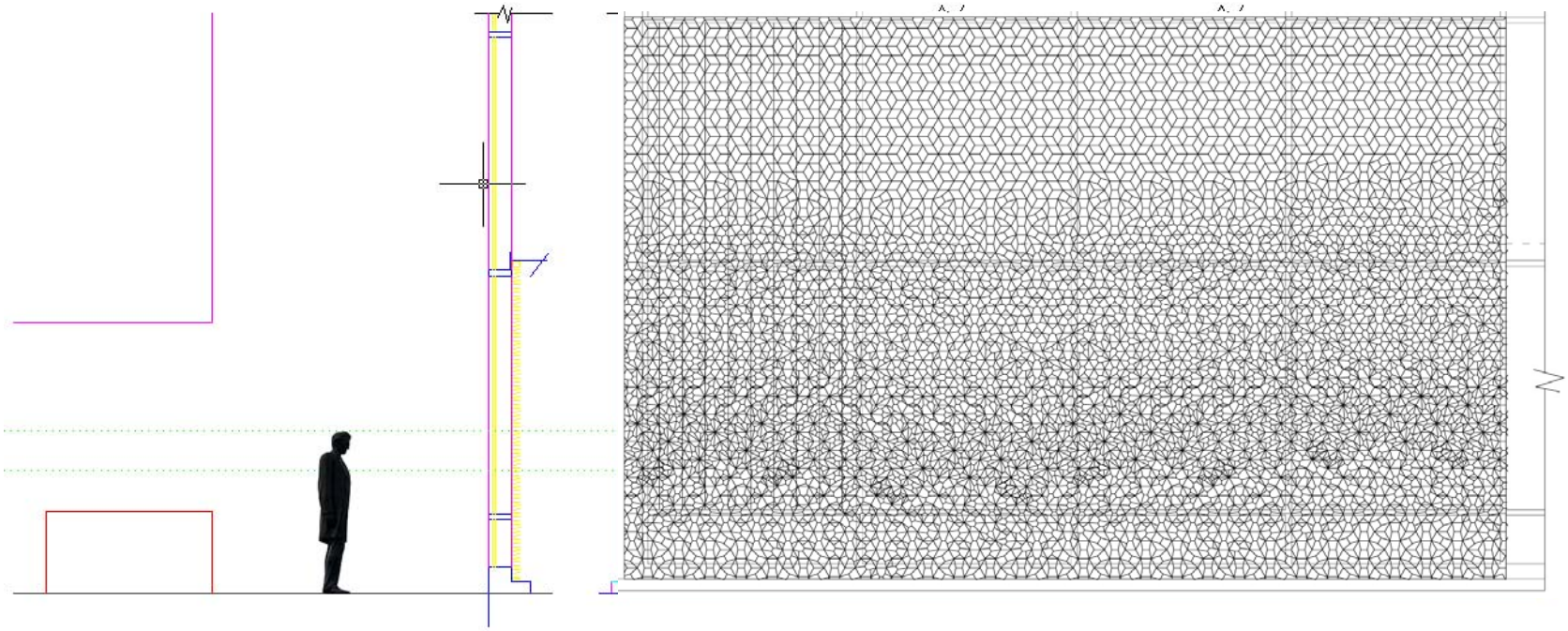


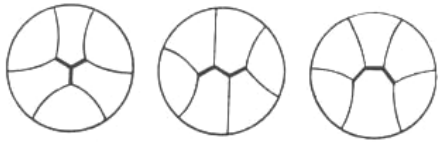




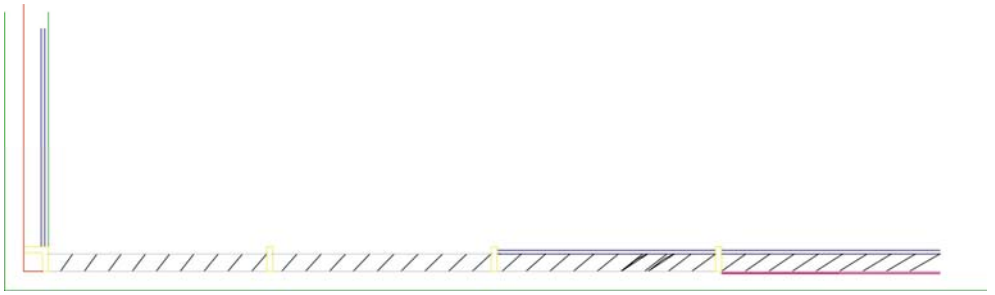


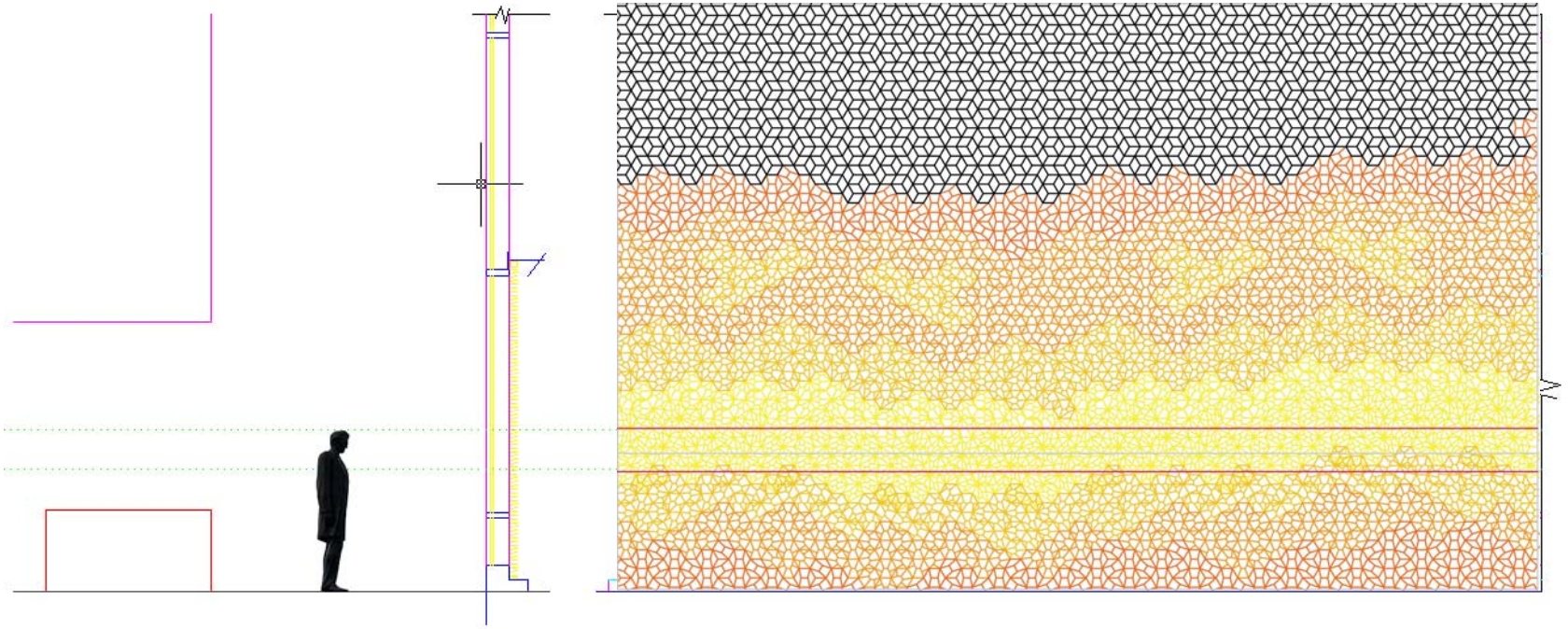
DESIGN STRATEGIES
cell division





DESIGN STRATEGIES
directing/diffusing screen





DESIGN STRATEGIES

scripting:

```
(defun slats (angle_1 angle_2 num_slats depth)

(command "ucs" "")
(command "osnap" "off")

(defun get_data_for_group_code (group_code entity / all_data pair_list the_data )
        (setq all_data  (entget entity))
        (setq pair_list (assoc group_code all_data))
        (setq the_data  (cdr pair_list))
)

(defun get_point (prompt / entity)
        (setq entity nil)
        (while (not (and (= (type entity) 'ENAME)
        (= (get_data_for_group_code 0 entity) "POINT")))
        (setq entity (car (entsel prompt))))
        entity)

(setq point1 (get_point "Pick first point"))
(terpri)
(setq point2 (get_point "Pick second point"))

(setq point1_coords (get_data_for_group_code 10 point1))
(setq point1_x (car point1_coords))
(setq point1_y (cadr point1_coords))

(setq point2_coords (get_data_for_group_code 10 point2))
(setq point2_x (car point2_coords))
(setq point2_y (cadr point2_coords))

(setq ttl_dist (distance (list point1_x point1_y 0.0) (list point2_x point2_y 0.0)))

(setq slat_num num_slats)
(setq slat_dist (/ ttl_dist slat_num))
(setq slat_incr slat_dist)

(setq angle_diff (abs (- angle_1 angle_2)))
(setq angle_incr (/ angle_diff num_slats))
(setq angle_step angle_1)

(repeat slat_num
        (setq x_1 (+ point1_x slat_incr))
        (setq angle_rad (* pi (/ angle_step 180.0)))
        (setq x_2 (- x_1 (* 20 (cos angle_rad))))
        (setq y_2 (- point2_y (* 20 (sin angle_rad))))
        (command "line" (list x_1 point1_y) (list x_2 y_2) "")
        (setq trim_point (get_data_for_group_code 11 (entlast)))
        (setq trim_list (cons trim_point trim_list))
        (setq slat_incr (+ slat_incr slat_dist))
        (setq angle_step (- angle_step angle_incr))
)
```

```
;(slats 48.5 42.0 9 7.25)

;

(defun slats (angle_1 angle_2 num_slats depth)

                (command "ucs" "")
                (command "osnap" "off")

                (defun get_data_for_group_code (group_code entity / all_data pair_list the_data )
                        (setq all_data  (entget entity))
                        (setq pair_list (assoc group_code all_data))
                        (setq the_data  (cdr pair_list))
                )

                (defun get_point (prompt / entity)
                        (setq entity nil)
                        (while (not (and (= (type entity) 'ENAME)
                        (= (get_data_for_group_code 0 entity) "POINT")))
                        (setq entity (car (entsel prompt))))
                        entity)

                (setq point1 (get_point "Pick first point"))
                (terpri)
                (setq point2 (get_point "Pick second point"))

                (setq point1_coords (get_data_for_group_code 10 point1))
                (setq point1_x (car point1_coords))
                (setq point1_y (cadr point1_coords))

                (setq point2_coords (get_data_for_group_code 10 point2))
                (setq point2_x (car point2_coords))
                (setq point2_y (cadr point2_coords))

                (setq ttl_dist (distance (list point1_x point1_y 0.0) (list point2_x point2_y 0.0)))

                (setq slat_num num_slats)
                (setq slat_dist (/ ttl_dist slat_num))
                (setq slat_incr slat_dist)

                (setq angle_diff (abs (- angle_1 angle_2)))
                (setq angle_incr (/ angle_diff num_slats))
                (setq angle_step angle_1)

                (repeat slat_num
                        (setq x_1 (+ point1_x slat_incr))
                        (setq angle_rad (* pi (/ angle_step 180.0)))
                        (setq x_2 (- x_1 (* 20 (cos angle_rad))))
                        (setq y_2 (- point2_y (* 20 (sin angle_rad))))
                        (command "line" (list x_1 point1_y) (list x_2 y_2) "")
                        (setq trim_point (get_data_for_group_code 11 (entlast)))
                        (setq trim_list (cons trim_point trim_list))
                        (setq slat_incr (+ slat_incr slat_dist))
                        (setq angle_step (- angle_step angle_incr))
                )
```

```
                    (setq ttl_dist (distance (list point1_x point1_y 0.0) (list point2_x point2_y 0.0)))

                    (setq slat_num num_slats)
                    (setq slat_dist (/ ttl_dist slat_num))
                    (setq slat_incr slat_dist)

                    (setq angle_diff (abs (- angle_1 angle_2)))
                    (setq angle_incr (/ angle_diff num_slats))
                    (setq angle_step angle_1)

                    (repeat slat_num
                                        (setq x_1 (+ point1_x slat_incr))
                                        (setq angle_rad (* pi (/ angle_step 180.0)))
                                        (setq x_2 (- x_1 (* 20 (cos angle_rad))))
                                        (setq y_2 (- point2_y (* 20 (sin angle_rad))))
                                        (command "line" (list x_1 point1_y) (list x_2 y_2) "")
                                        (setq trim_point (get_data_for_group_code 11 (entlast)))
                                        (setq trim_list (cons trim_point trim_list))
                                        (setq slat_incr (+ slat_incr slat_dist))
                                        (setq angle_step (- angle_step angle_incr))
                    )
                    (setq trim_list2 trim_list)

                    (setq offst_dist depth)
                    (setq slat_incr slat_dist)
                    (command "line" point1_coords point2_coords "")
                    (setq trim_line (entlast))
                    (command "offset" offst_dist trim_line (list point1_x (- point1_y 1)) "")
                    (setq trim_line2 (entlast))
                    (setq trim_num 0)
                    (repeat slat_num
                                        (setq trim_point2 (nth trim_num trim_list2))
                                        (setq trimx (car trim_point2))
                                        (setq trimy (cadr trim_point2))
                                        (command "zoom" "c" (list trimx trimy 0) 0.25)
                                        (command "trim" trim_line2 "" (list trimx trimy) "")
                                        (setq trim_num (+ 1 trim_num))
                    )

                    (setq trim_list nil)
)

;(slats 55.0 48.5 9 7.25)

;(slats 48.5 42.0 9 7.25)

;(slats 42.0 35.5 9 7.25)

(slats 35.5 29.0 9 7.25)
```