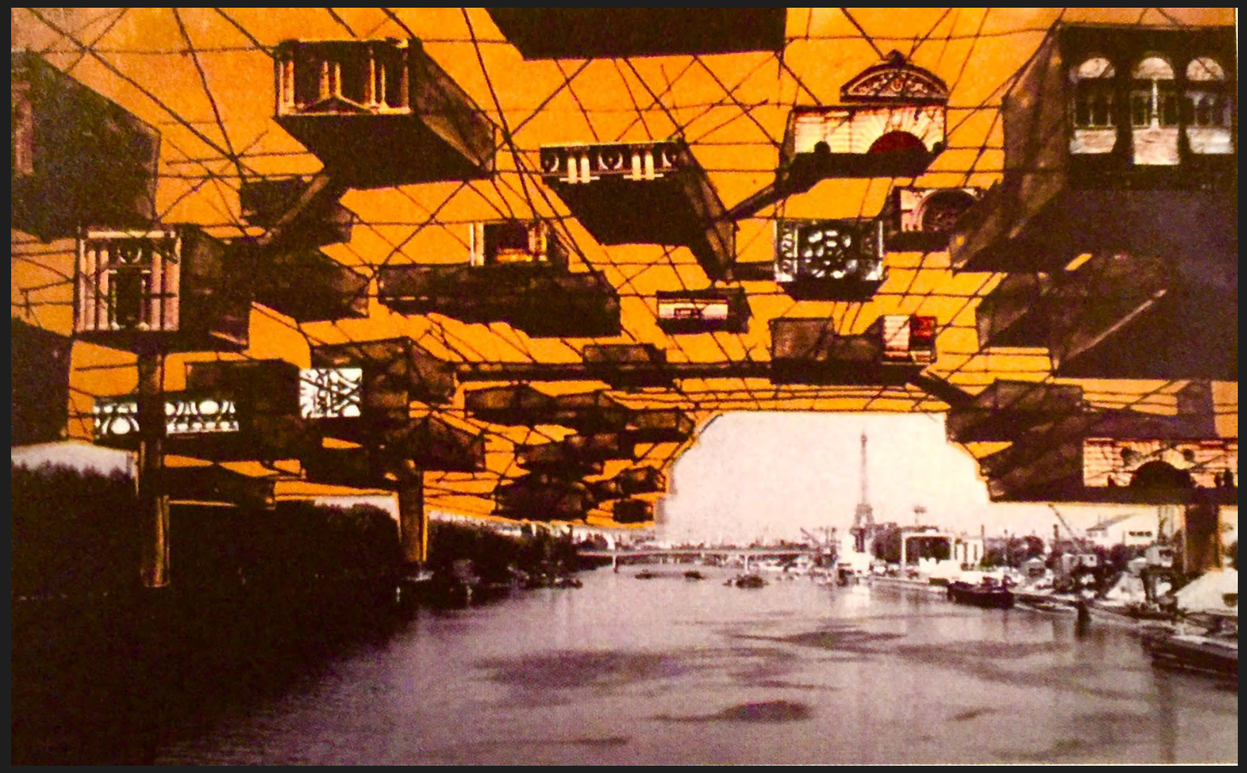
ROBOTIC ARCHITECTURAL SPECIES

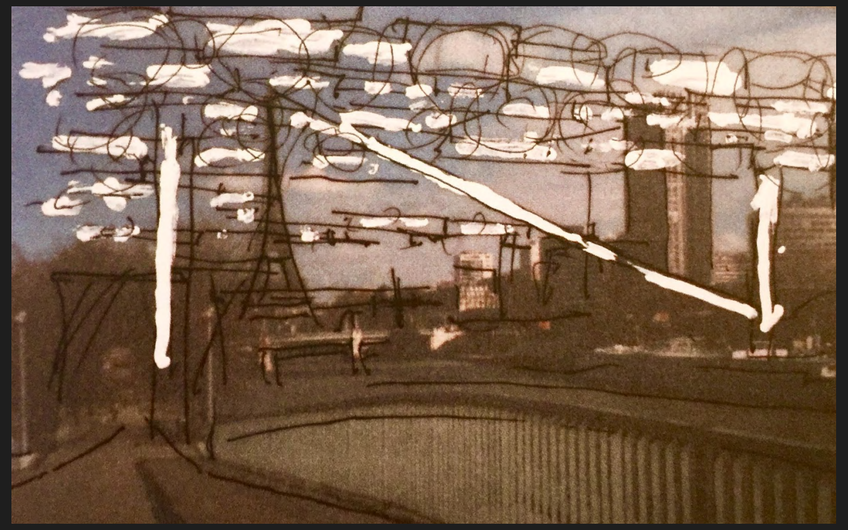
DYNAMIC ARCHITECTURAL AND INTERACTIVE SYSTEM: AGGREGATE STRUCTURES AND SWARM ROBOTICS

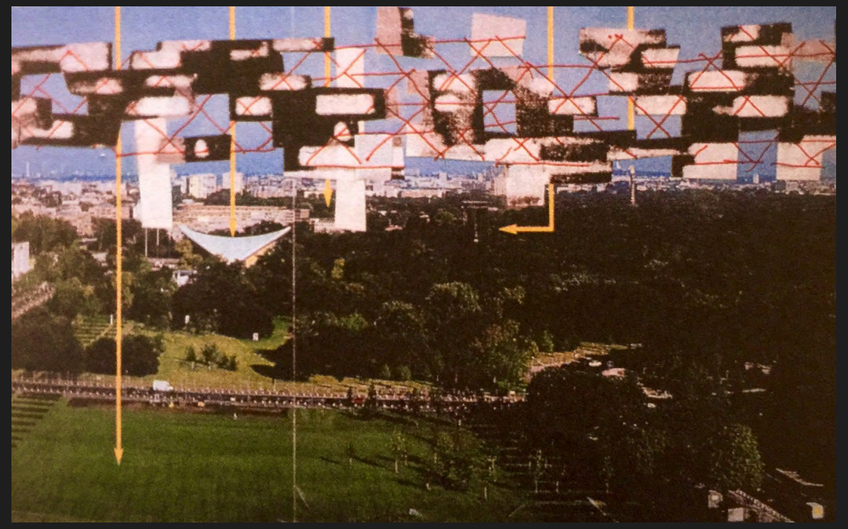
Thierry Syriani / 07-07-2020

CONTEXT

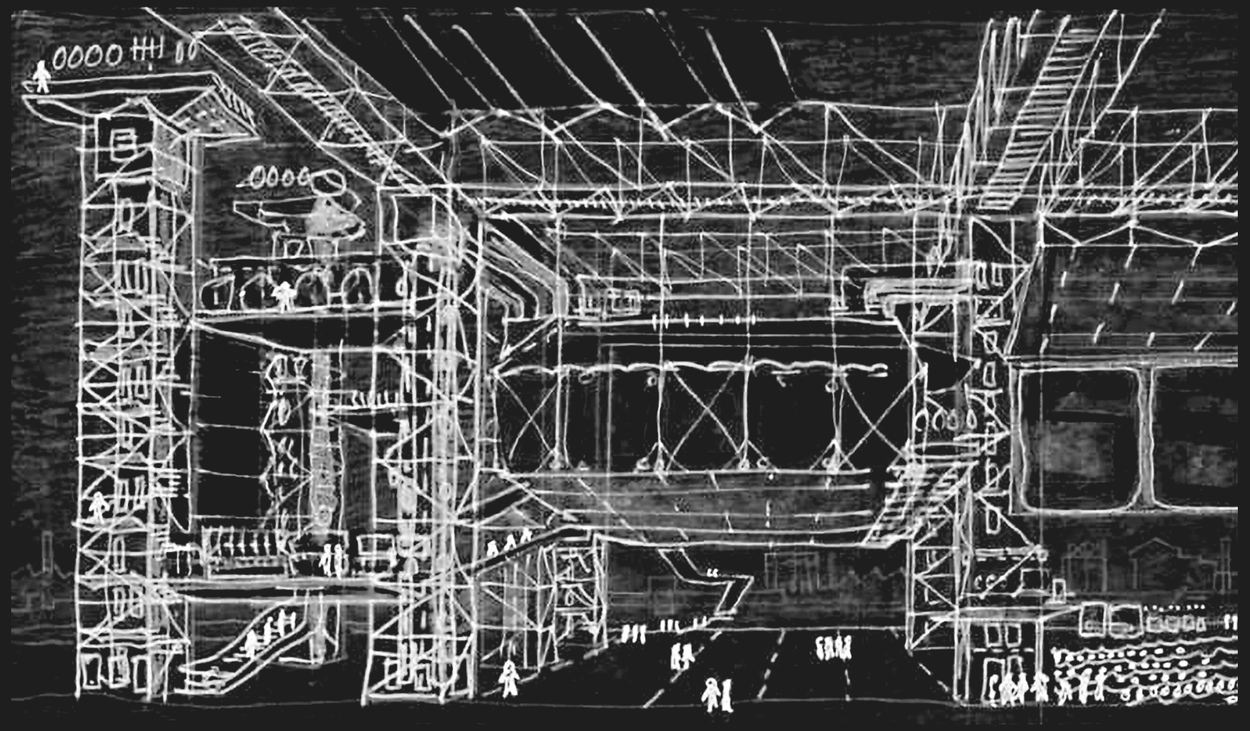
SOFTNESS + ROBOTICS

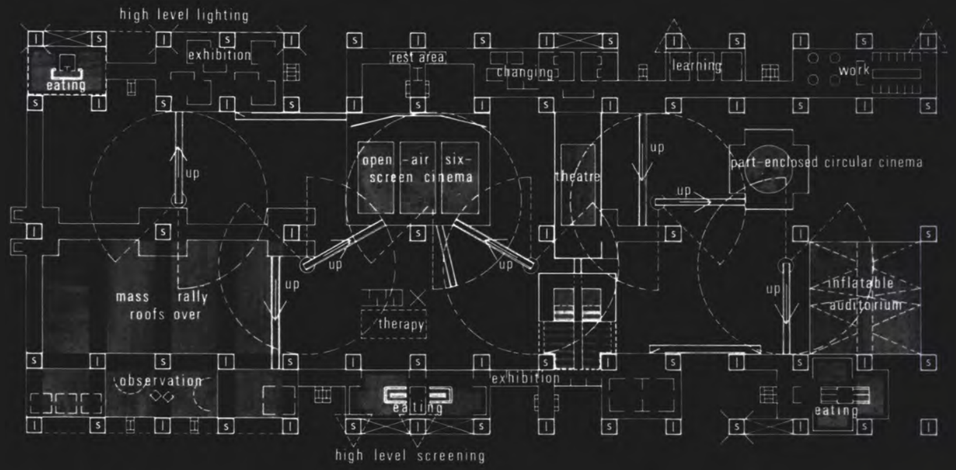


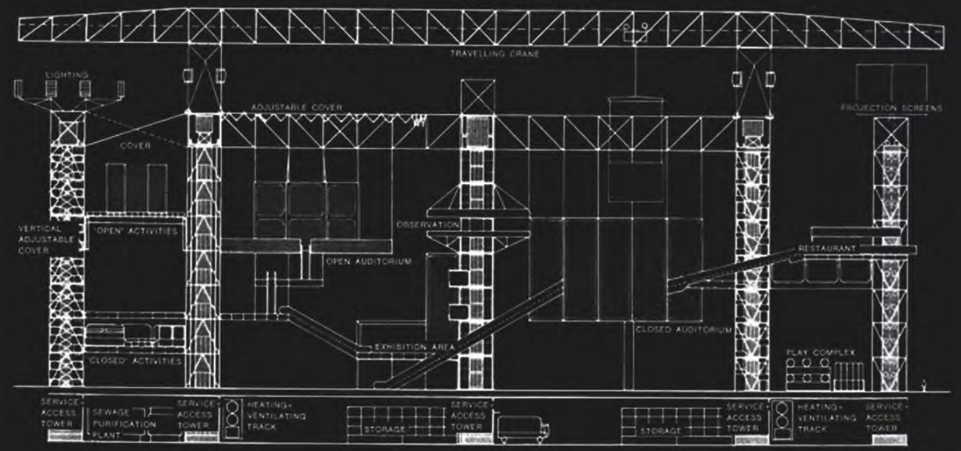




SPATIAL CITY, YONA FRIEDMAN, 1960



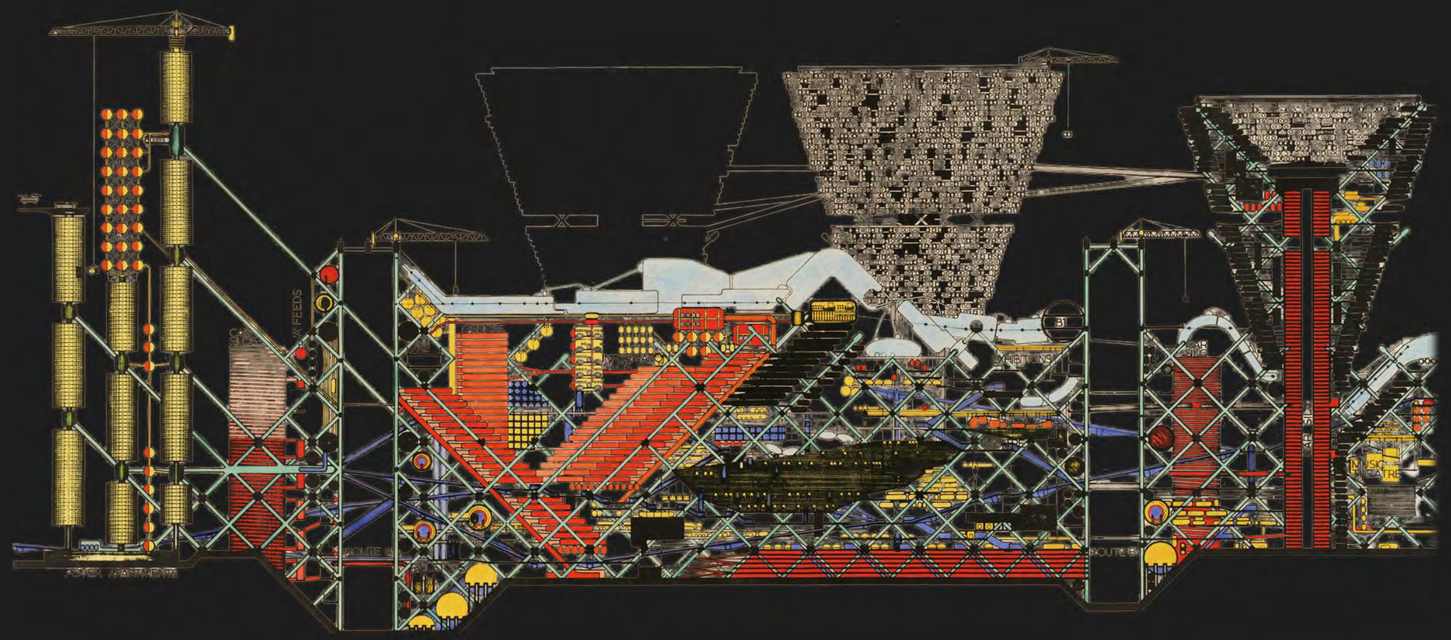


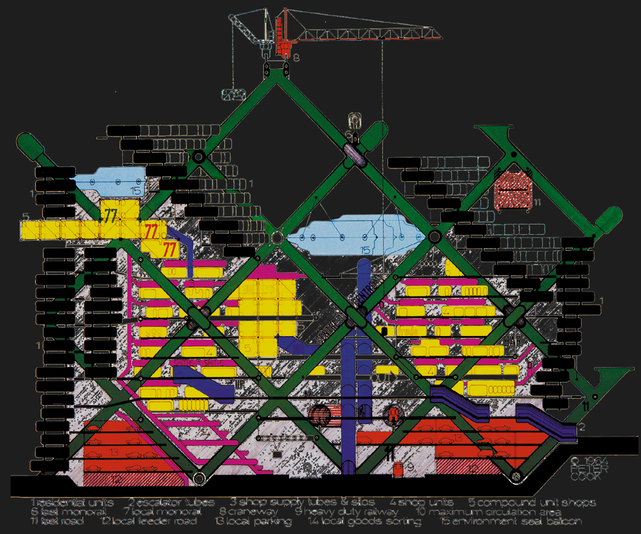


CEDRIC PRICE, FUN PALACE, 1961

“Wonder is very tricky to design for, since it can usually be experienced only once.”

Cedric Price, 1966



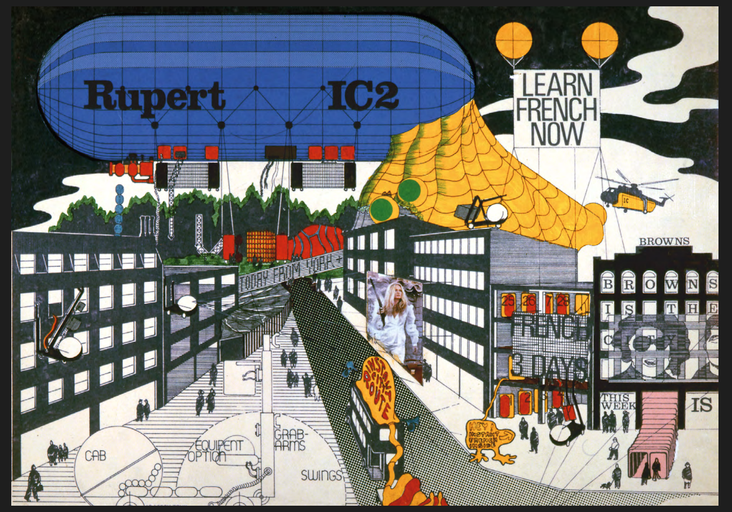


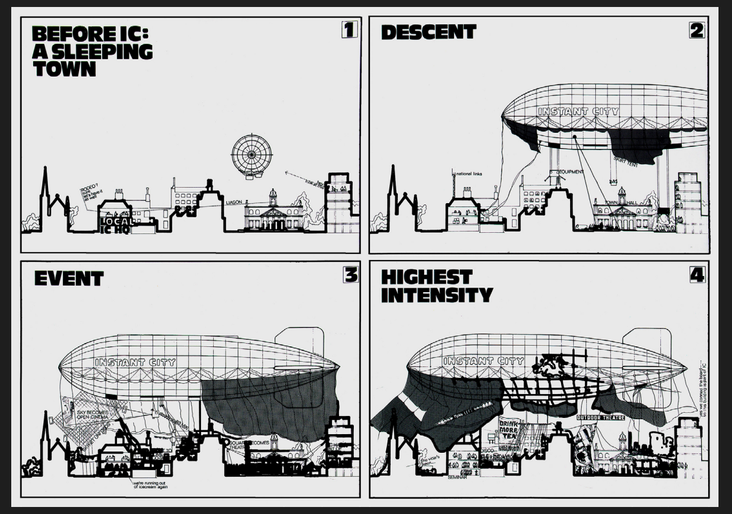
PLUG-IN CITY, ARCHIGRAM, 1964

“An architecture of incompleteness”

Simon Saddler
“Archigram: Architecture without Architecture”, 2005







INSTANT CITY, ARCHIGRAM, 1964



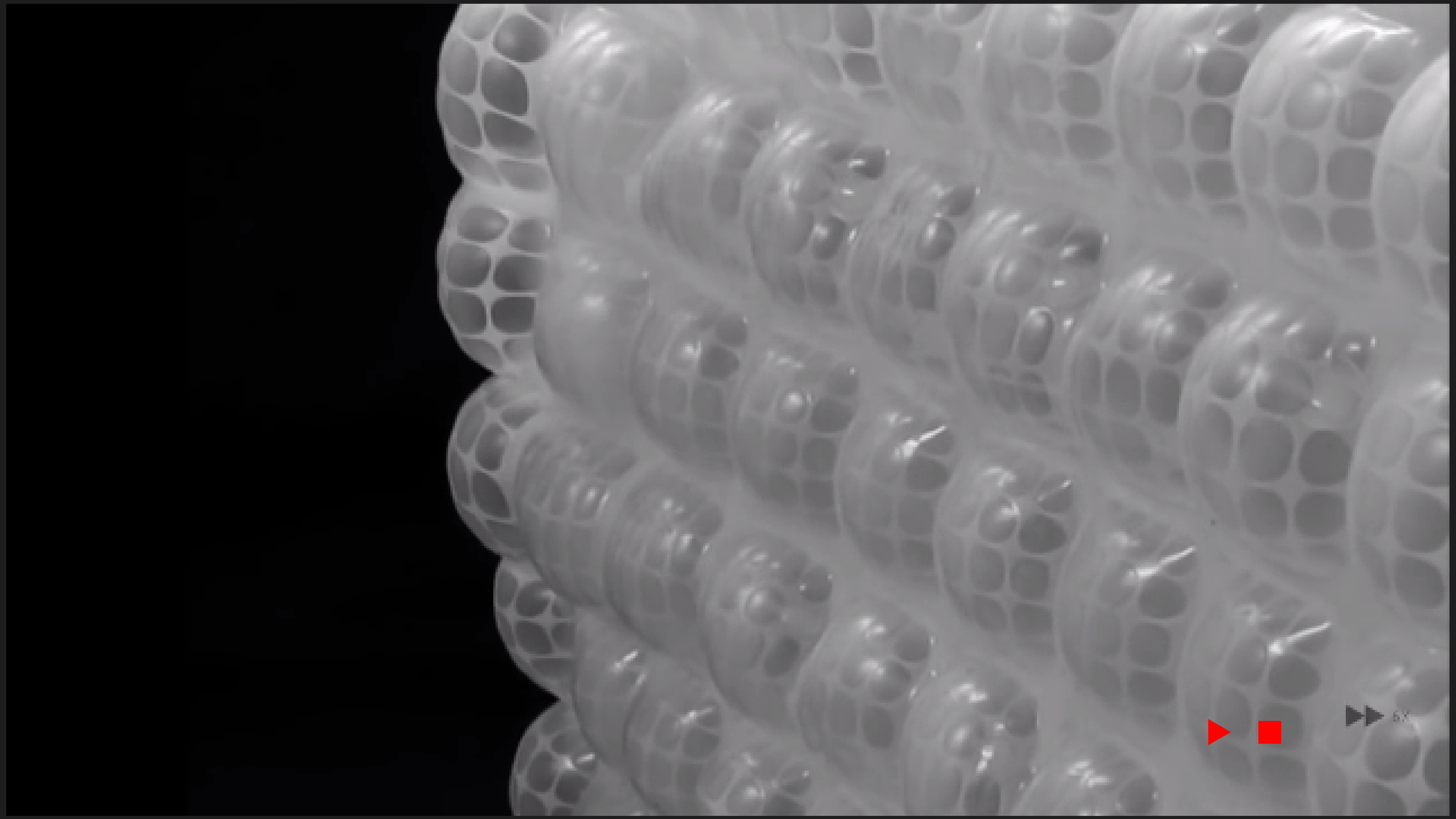
SILVER CLOUDS, ANDY WARHOL, 1966



BLOW CHAIR, ZANOTTA, 1967



INFLATABLE SILICONE MUSCLE, BARTLETT UNIVERSITY, UK



SWARM INTELLIGENCE AND THE GENERATIVE ALGORITHM

```
ues = true;
ar = false;

P2D);
rs();
PVec-
ght/2);

55,100);

ock();
tial set of
system
0; i < 10;

d(new
ight/2));

new Array-
);
0; i < obsNum;

dd(new Ob-
width), ran-

5);
acles);
rs();
0; i < obsta-
+){
= obstacles.
;

ssed &&
```

```
    flock.addBoid(new Boid(-
mouseX,mouseY));
  }


  if (showvalues) {
    fill(0);
    textAlign(LEFT);
    text("Total boids: " +
flock.boids.size() + "\n" +
"Framerate: " + round(fram-
eRate) + "\nPress any key to
show/hide sliders and text\
nClick mouse to add more
boids",5,200);
  }
}

void keyPressed() {
  showvalues = !showvalues;
}

void mousePressed() {
}

----------------

// The Nature of Code
// Daniel Shiffman
// http://natureofcode.com

float swt = 25.0;
float awt = 4.0;
float cwt = 5.0;
float collisionwt = 5;
float wanwt = 2;
float owt = 2;

float maxspeed = 1;
float maxforce = 0.025;

float SS = 50.0;
float AS = 50.0;
float CS = 50.0;

float CollisionStrength = 0.5;
float avoidStrength = 3.0;

float projectX, projectY;
float safetyRad = 10.0;
int checkLength = 100;
```

```
PShape triangle;
PShape separation;
PShape cohesion;

ArrayList<Obstacle> obsta-
cles;

boolean collisionDetected;
color boidColor = 0;

// Flocking
// Daniel Shiffman <http://
www.shiffman.net>
// The Nature of Code, Spring
2009

// Boid class
// Methods for Separation,
Cohesion, Alignment added

class Boid {

  PVector pos;
  PVector vel;
  PVector acc;
  float r;

  boolean stuck;
  boolean isIntersecting;

  PShape obstacleVision;
  float wandertheta;

  Boid(float x, float y) {
    acc = new PVector(0,0);
    vel = new PVector(ran-
dom(-1,1),random(-1,1));
    pos = new PVector(x,y);
    r = 6.0;
    obstacleVision = create-
Shape();
    wandertheta = 0;
  }

  void run(ArrayList<Boid>
boids) {
    flock(boids);
    update();
    borders();
    /*collision();*/
```

```
    avoidVis();
    wanderVis();
    render();
  }

  ////////////////////////////
////////////
  //
  // APPLY SINGLE BEHAVIOR
  void applyForce(PVector
force) {
    // We could add mass here
if we want A = F / M
    acc.add(force);
  }

  ////////////////////////////
////////////
  //
  // APPLY ALL BEHAVIORS
  void flock(ArrayList<Boid>
boids) {
    PVector sep = separate(
rate(boids);
    PVector ali =
ment(boids);
    PVector coh = cohe-
sion(boids);   // Cohesion
    PVector collision = col-
lision(boids); //Collision
    PVector wan = wander();
// Wander
    PVector obs = avoid(); //
Avoid obstacle

    sep.mult(swt); // Arbi-
trarily weight these forces
    ali.mult(awt);
    coh.mult(cwt);
    collision.mult(collision-
wt);
    wan.mult(wanwt);
    obs.mult(owt);

    applyForce(obs);
    applyForce(wan);
    applyForce(sep); // Add
the force vectors to acceler-
ation
    applyForce(ali);
```

```
    sep.mult(swt); // Arbi-
trarily weight these forces
    ali.mult(awt);
    coh.mult(cwt);
    collision.mult(collision-
wt);
    wan.mult(wanwt);
    obs.mult(owt);

    applyForce(obs);
    applyForce(wan);
    applyForce(sep); // Add
the force vectors to acceler-
ation
    applyForce(ali);
    applyForce(coh);
    applyForce(collision);

  }

  ////////////////////////////
////////////
  //
  // UPDATE
  void update() {
    vel.add(acc); // Update
velocity
    vel.limit(maxspeed); //
Limit speed
    pos.add(vel); // Update
position
    acc.mult(0); // Reset ac-
celertion to 0 each cycle
  }

  void avoidForce(){

    //FIND FUTURE POS
    PVector forward = vel.
get();
    PVector predict = for-
ward.normalize();
    predict.mult(check-
Length);
    PVector predictpos =
PVector.add(pos, predict);

    //FIND CLOSEST OBSTACLE
TO FUTURE POS
```

```
    float closestDist =
1000000;
    int closestObs = 0;
    int touchedObs = 0;
    float minDist2 = radius +
safetyRad;

    for (int j=0; j < obsta-
cles.size(); j++){
      PVector obsPos = obsta-
cles.get(j).position.copy();
      float minDist = obsta-
cles.get(j).radius;
      float actualDist = PVec-
tor.dist(predictpos,obsPos);

      if(actualDist < clos-
estDist){
        closestDist = actu-
alDist;
        closestObs = j;
      }
    }

    boolean collisionDetected
= false;
    PVector steer = vel.
get();

    //CHECK FOR COLLISION
    if (closestDist <
minDist2) {
      touchedObs++;
      collisionDetected =
true;
      //REACT FORCE: MOVE
AWAY
      steer = avoid();
    }
    else {
      collisionDetected =
false;
      //REACT FORCE: KEEP
WANDERING
      steer = wander();
    }

    steer.limit(maxforce);
    applyForce(steer);

}
```

```
  ////////////////////////////
////////////
  //
  //AVOID OBSTACLE STEERING
  PVector avoid(){

    PVector forward = vel.
get();
    PVector predict = for-
ward.normalize();
    predict.mult(check-
Length);
    PVector predictpos =
PVector.add(pos, predict);

    float closestDist =
1000000;
    int closestObs = 0;
    int touchedObs = 0;
    PVector steer = vel.
get();

    for (int j=0; j < obsta-
cles.size(); j++){
      PVector obsPos = obsta-
cles.get(j).position.copy();
      float minDist = obsta-
cles.get(j).radius;
      float actualDist = PVec-
tor.dist(predictpos,obsPos);

      if(actualDist < minDist
&& actualDist < closestDist){
        touchedObs++;
        closestDist = actu-
alDist;
        closestObs = j;
      }
    }

    //PVECTOR ARRAY FOR
SMOOTH OBSTACLE AVOIDANCE
    PVector steerZero = new
PVector();
    PVector steerOne = steer-
Zero.copy();
    PVector steerTwo =
steerOne.copy();
    PVector steerThree =
steerTwo.copy();
```

```
PVe
steerTh

  //C
VECTOR
  PVe
steer.c

  //A
FIRST P
  ste

  //A
OF LAST
  ste
d(steer
Two);
  ste
d(steer
  ste

  if
    P
add(pre
get(clo
    fl
gleBetw
    P
tion =
vel.x,0
  c
div(dis
  c
mult(av

  //C
rection
with ob
"lefty"
  i
Temp.ad
// Move
  e
(choose
left
  }

  ste

  ret
```



STARLING MURMURATION



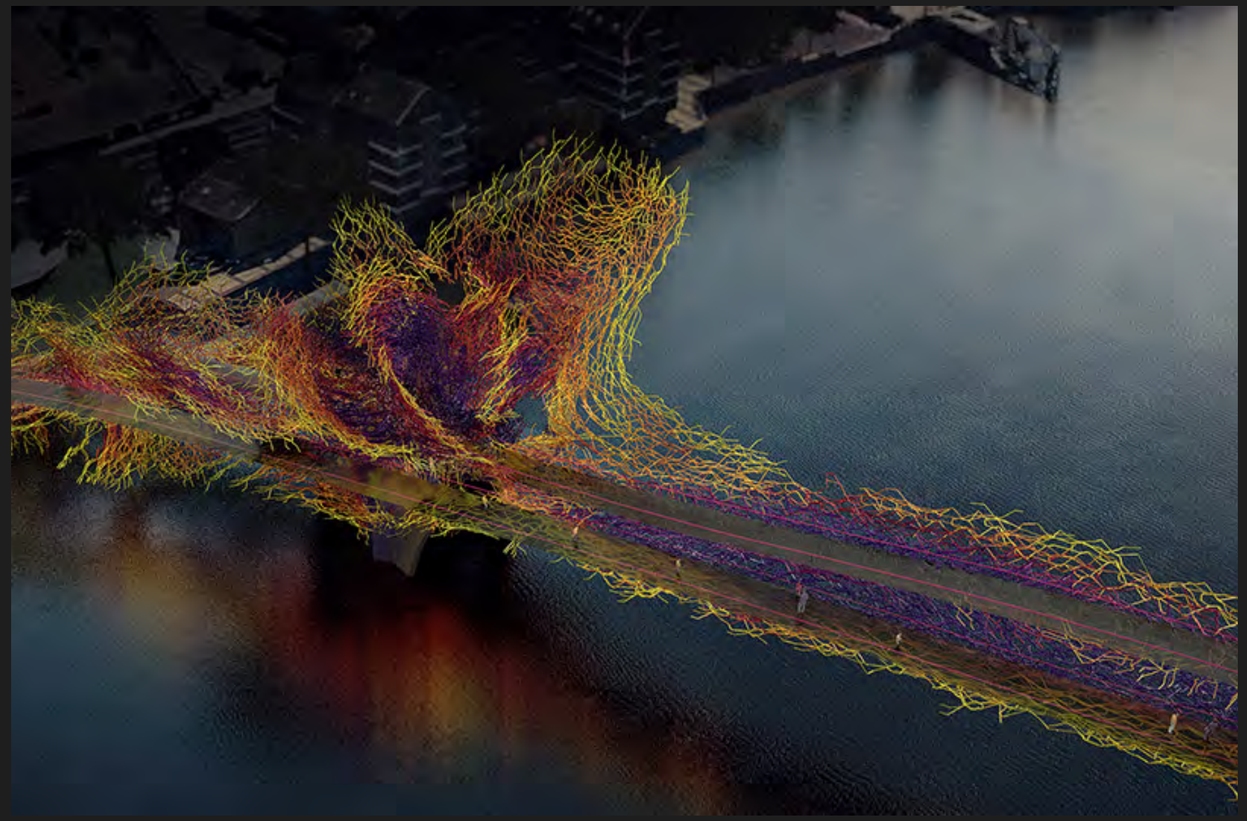
STARLING MURMURATION



STARLING MURMURATION

“This is a shift in focus from designing the form of the object to designing the nature of the object.”

Roland Snooks



NINE ELMS BRIDGE, LONDON, 2015

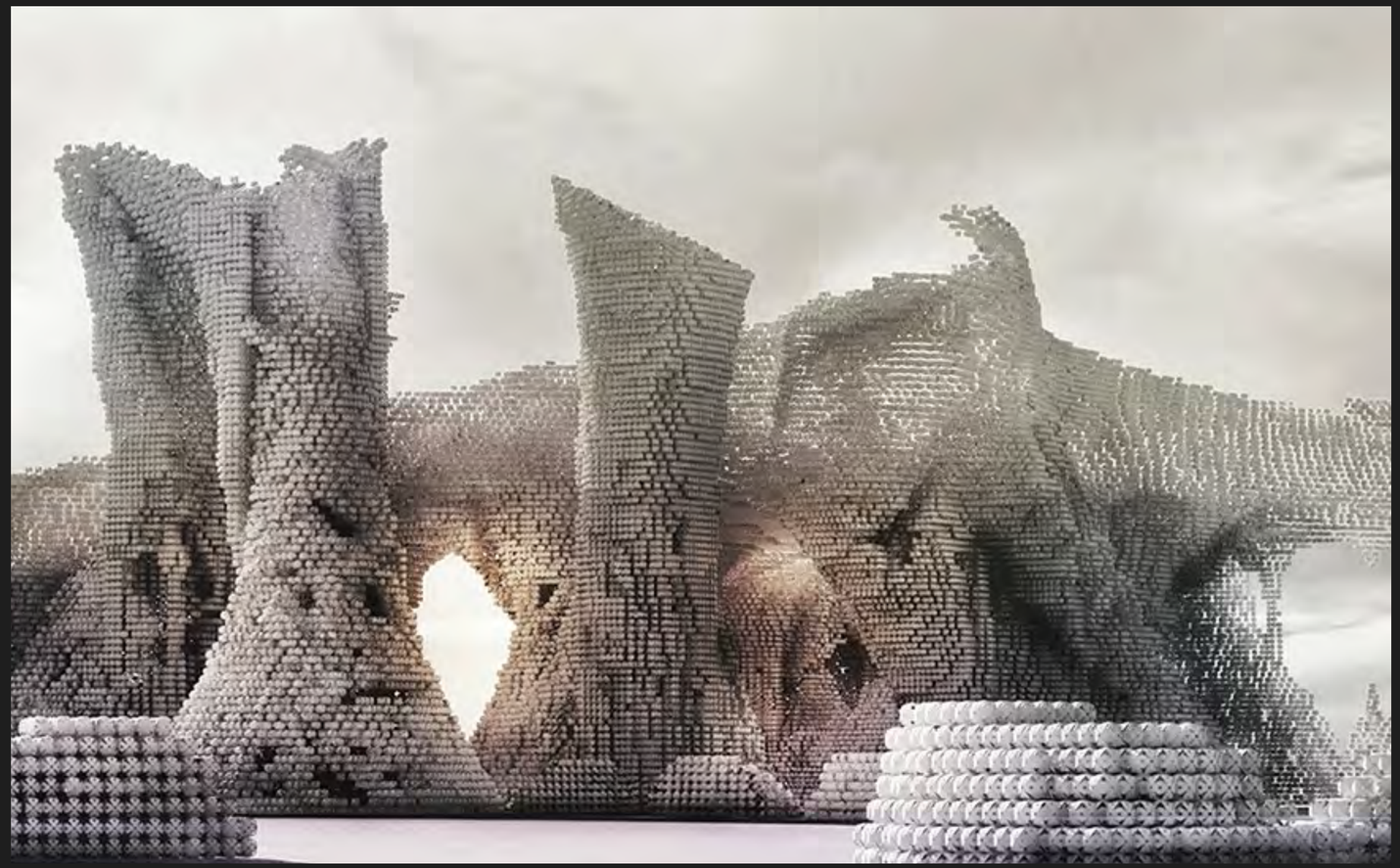


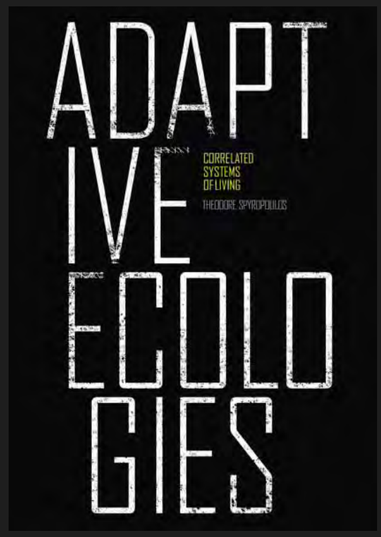
BRASS SWARM, SHANGHAI, 2015



HAWTHORN PUBLIC ART, MELBOURNE, 2016

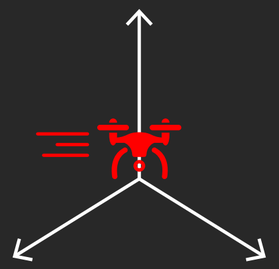
ROLAND SNOOKS, SWARM ALGORITHM



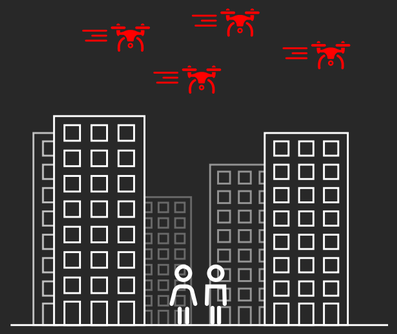




HYPERCELL, AADRL, LONDON, 2013



FAST, FLUID, FREE
MOVEMENTS IN 3D SPACE



LESS INTRUSIVE
IN AN URBAN CONTEXT



AUTONOMOUS AND
SWARMING BEHAVIORS

UAVS IN ARCHITECTURE



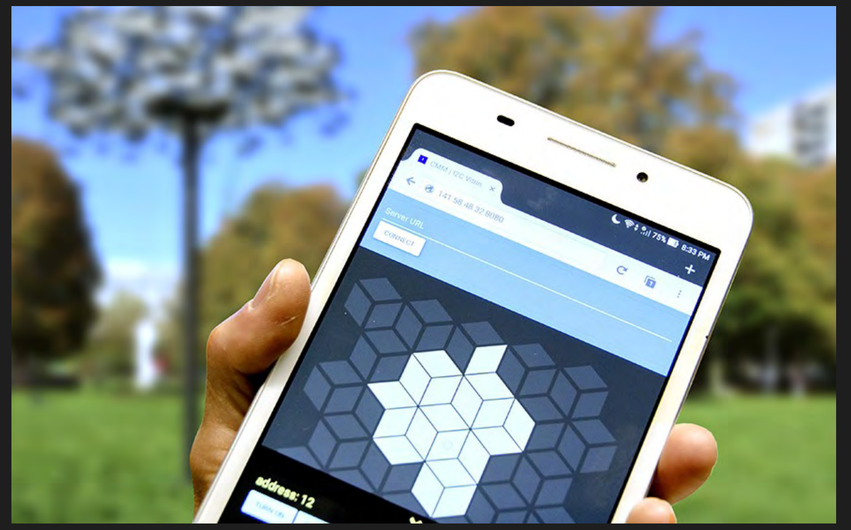




FLIGHT ASSEMBLED ARCHITECTURE GRAMAZIO+KOHLER, 1964







INSTANT CITY, ARCHIGRAM, 1964

DYNAMIC ARCHITECTURAL SYSTEM: DESIGNING A “SPECIES”

ARCHITECTURAL CONCEPT

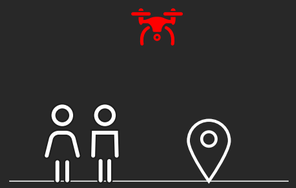


FORM TEMPORARY SHELTERS



RESPOND TO CLIMATE CONDITIONS

INTERACTION CONCEPT



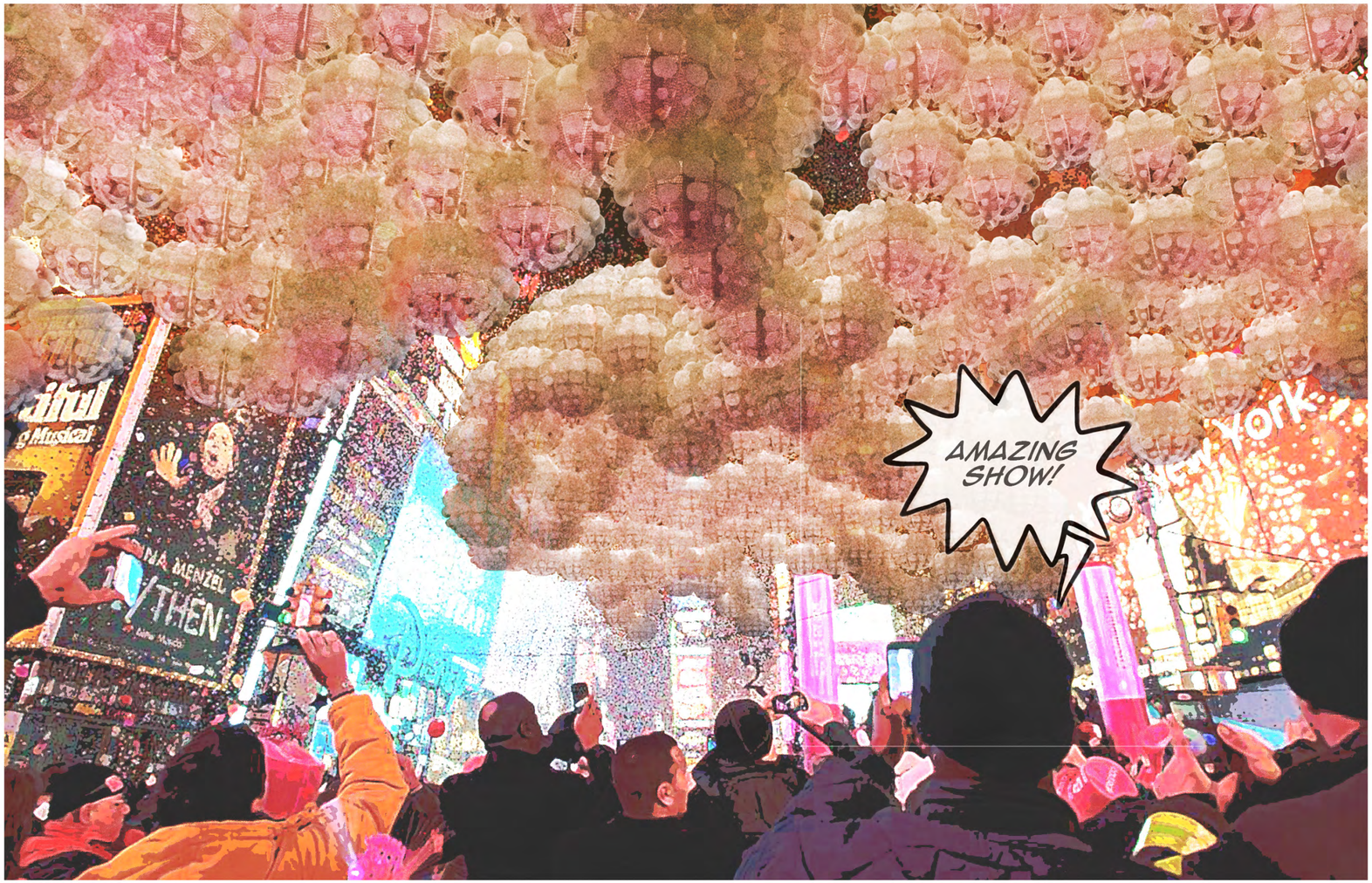
USER INTERACTION

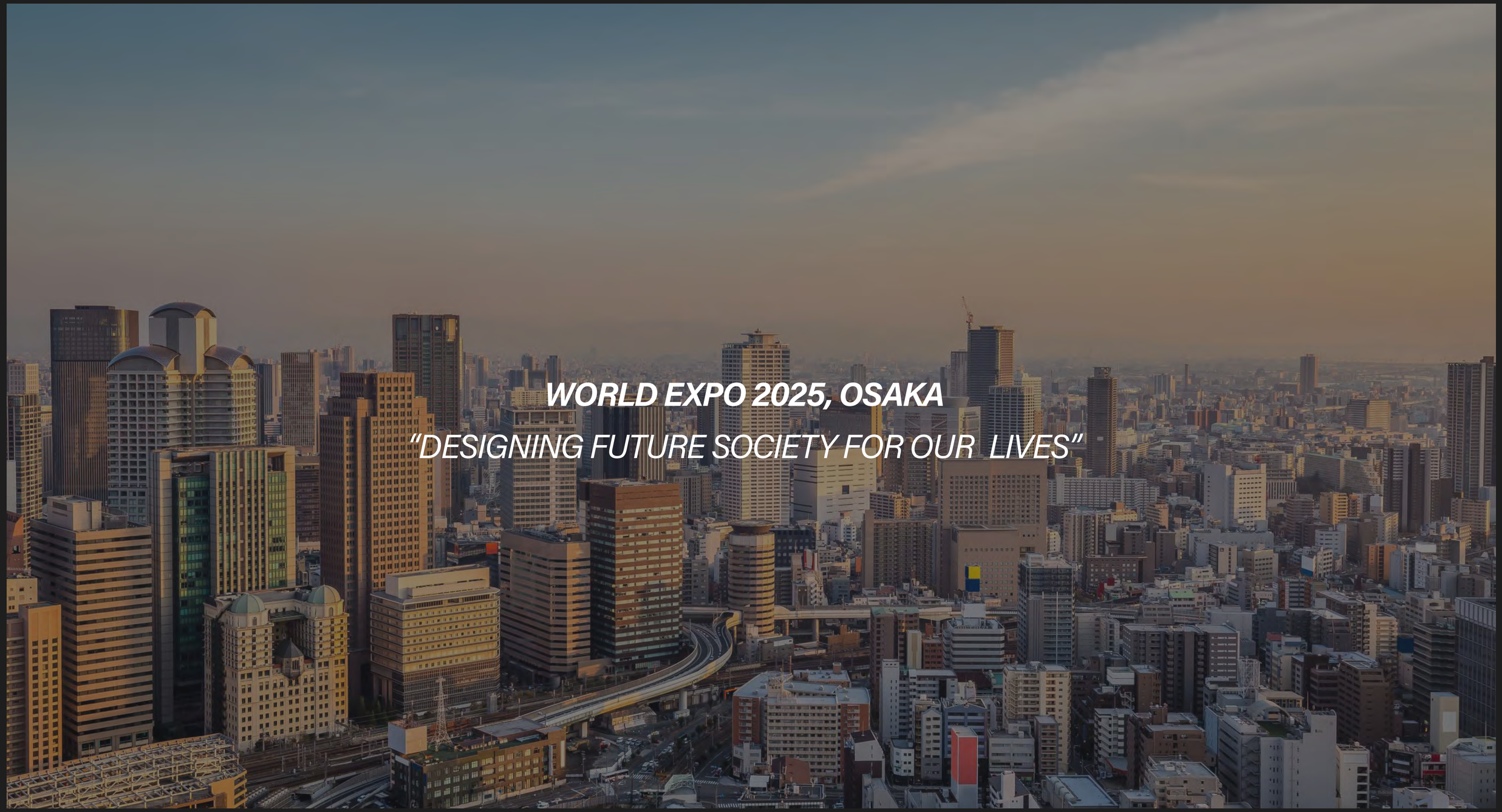


PLAYFULNESS AND DELIGHT


WOOHOO!
I LOVE THIS CONCERT!

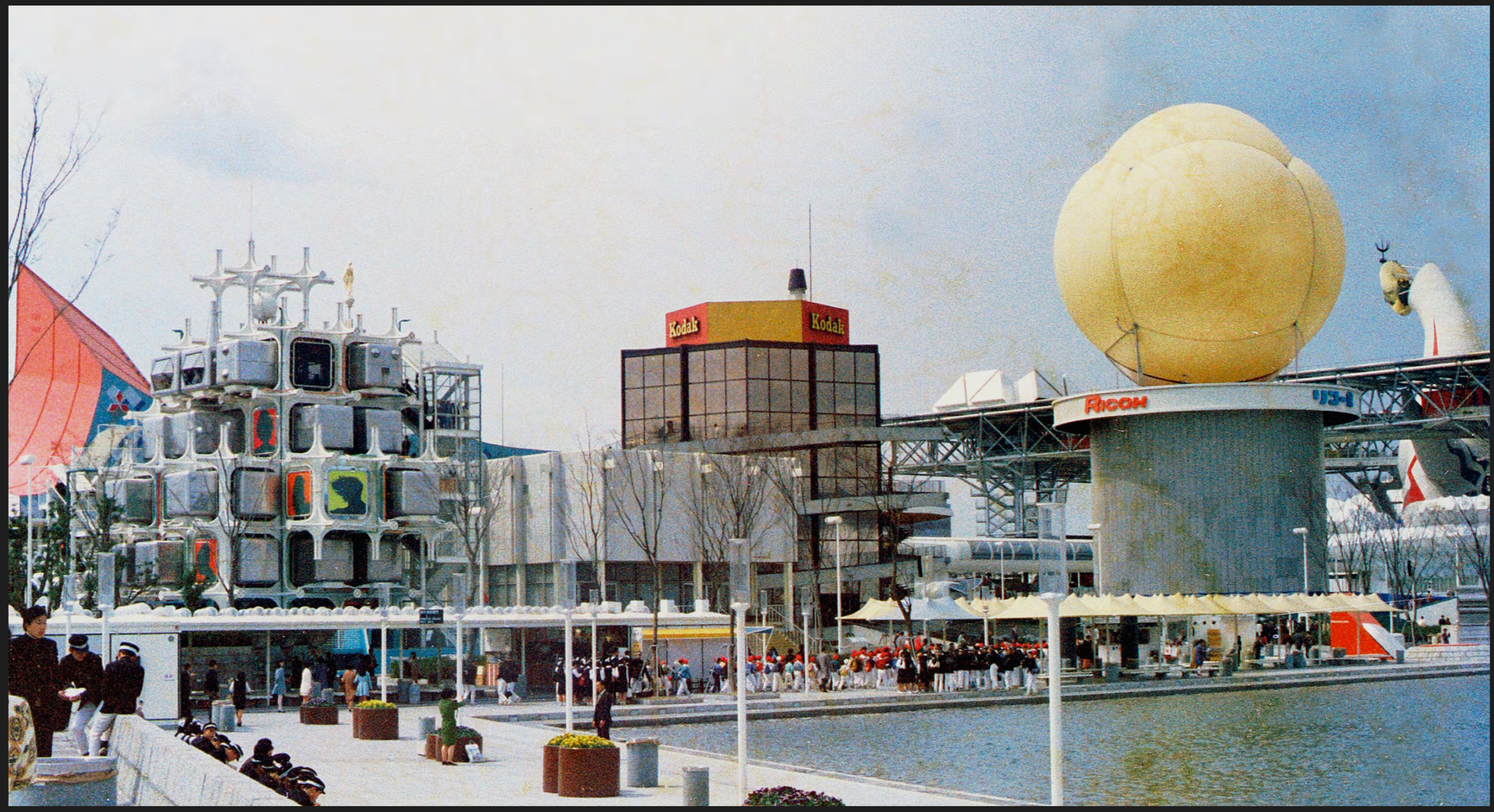

I'M SINGING IN THE RAIN!
OPEN 24 HOURS
A BRONX TALE
NASHVILLE
TOSHIBA
MARRIOTT MARQUIS
"ANDREW LLOYD WEBBER HAS BROADWAY ROCKING!"
Are you ready to wake up?
GREAT COMET
OUTERBORO
AMERICAN EAGLE OUTFITTERS
20 YEARS!
CHICAGOTHEMUSICAL.COM
McDonald's


AMAZING SHOW!

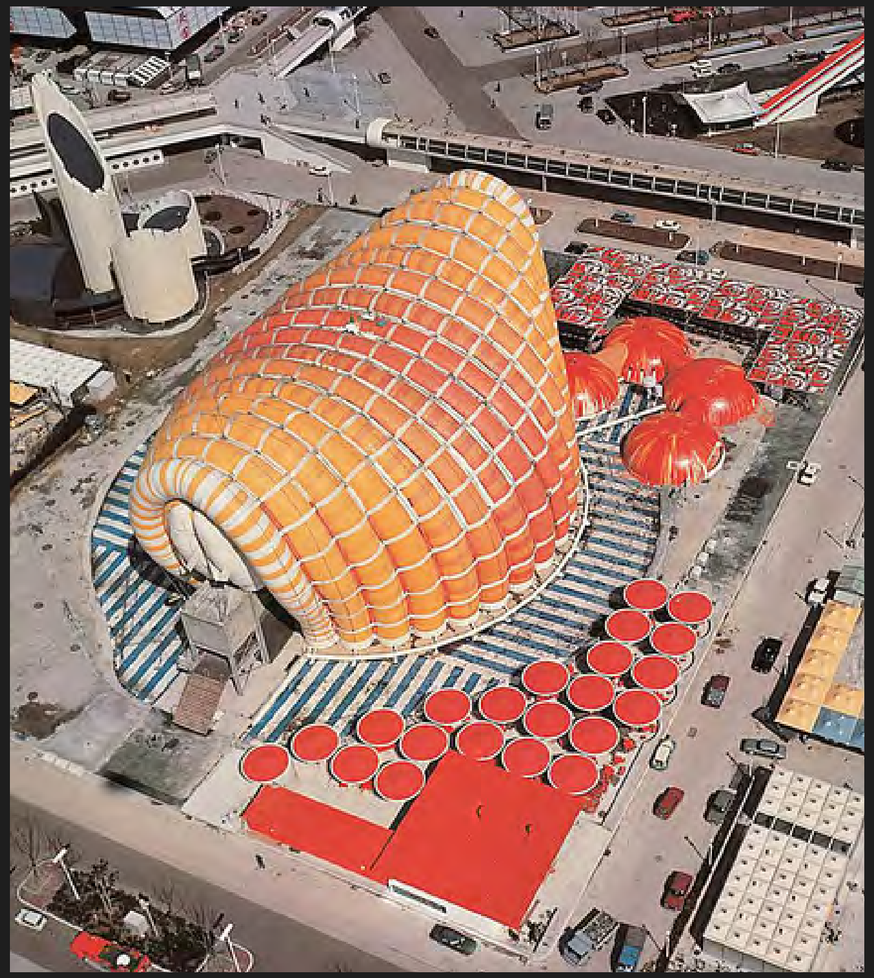

WORLD EXPO 2025, OSAKA
“DESIGNING FUTURE SOCIETY FOR OUR LIVES”



OSAKA, WORLD EXPO, 1970



OSAKA, WORLD EXPO, 1970



FUJI PAVILION BY YUKATA MURAKA



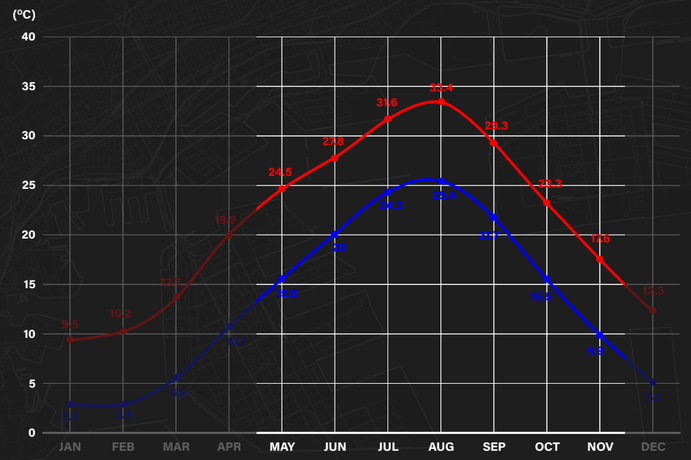
OSAKA, WORLD EXPO, 1970

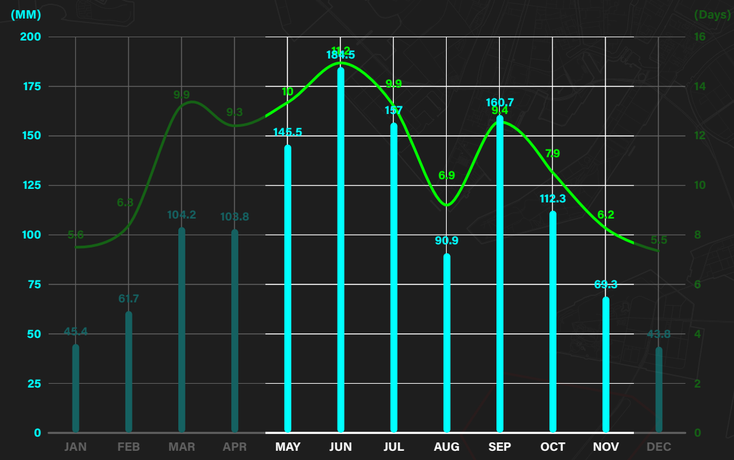


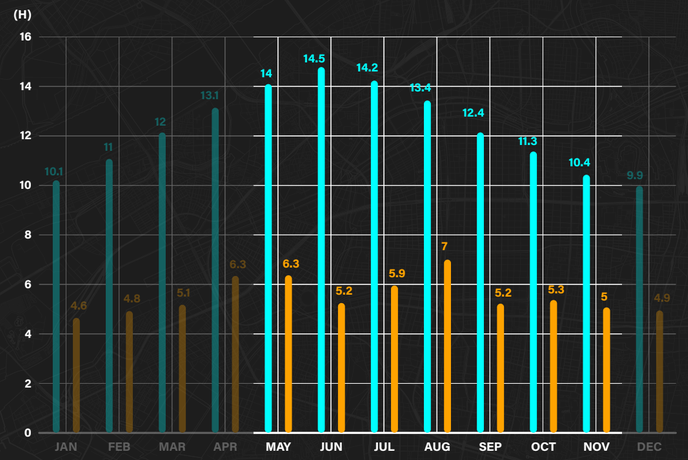
US PAVILLION, DAVIS BRODY BOND

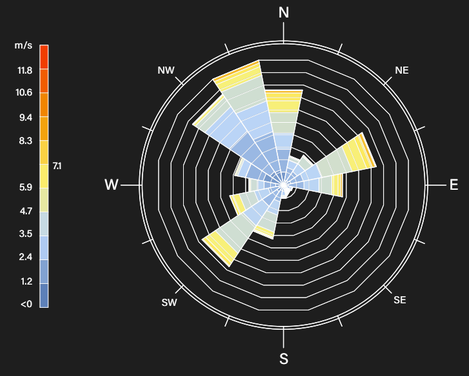


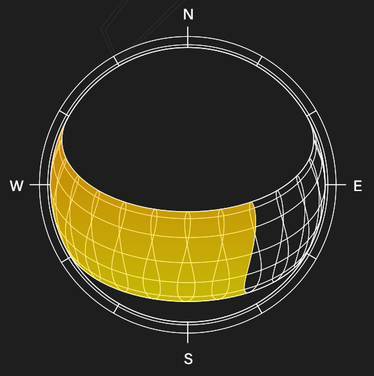
OSAKA, WORLD EXPO, 1970

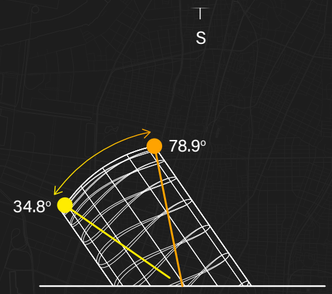

(°C)
40
35
30
25
20
15
10
5
0
JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC

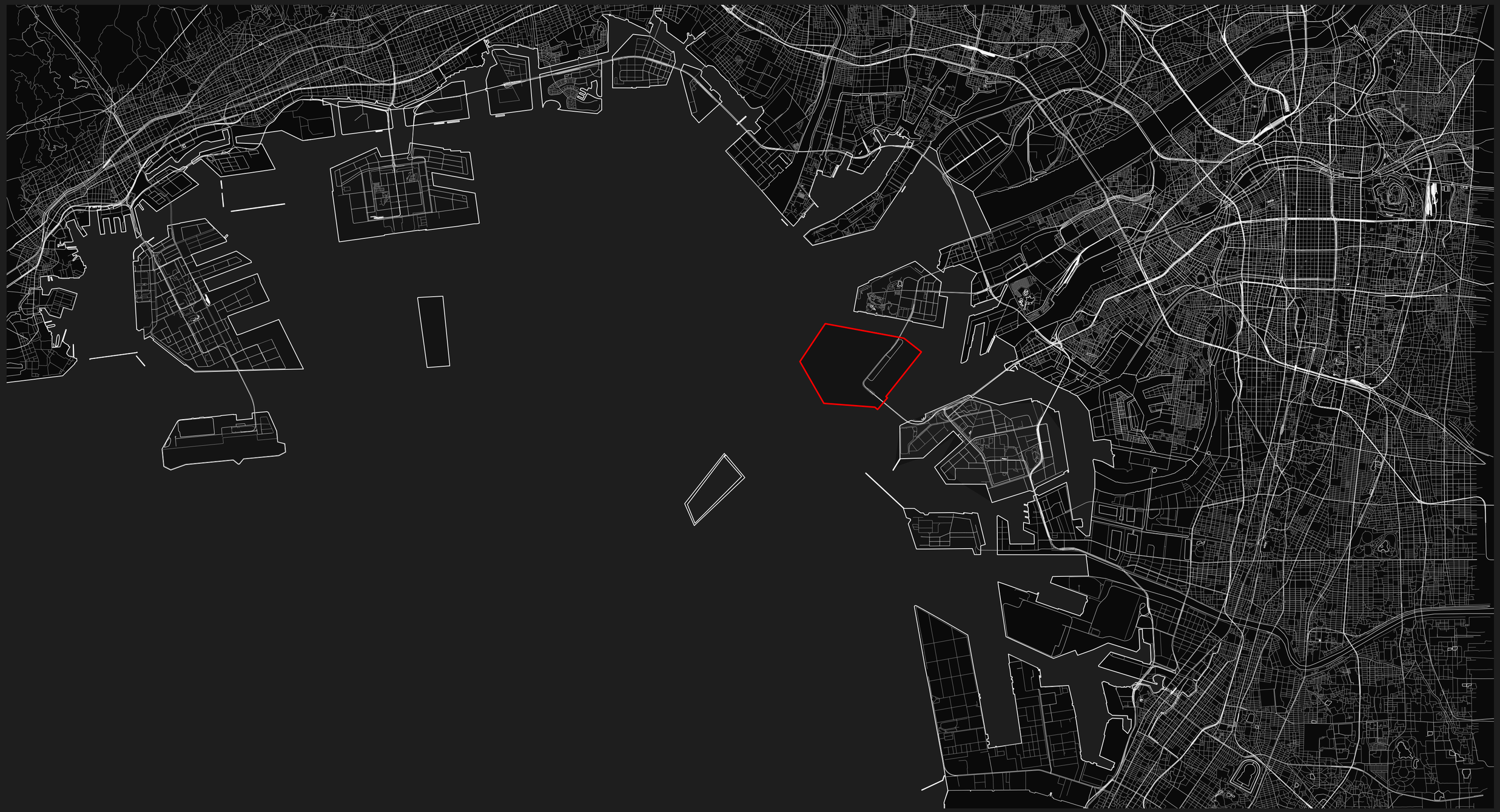

(MM)
(Days)
200
175
150
125
100
75
50
25
0
JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC

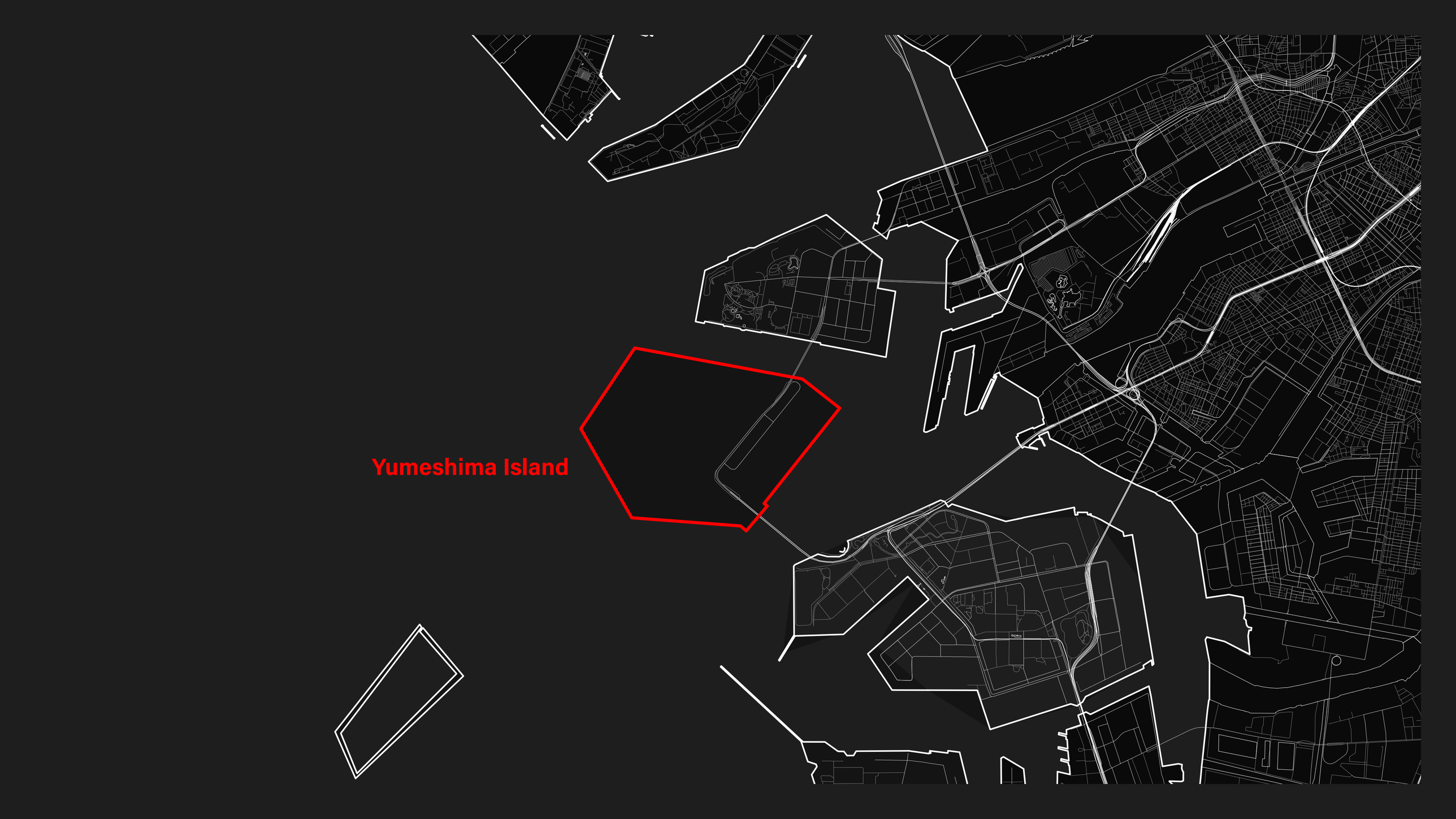

(H)
16
14
12
10
8
6
4
2
0
JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC


m/s
11.8
10.6
9.4
8.3
7.1
5.9
4.7
3.5
2.4
1.2
<0
N
NE
E
SE
S
SW
W
NW

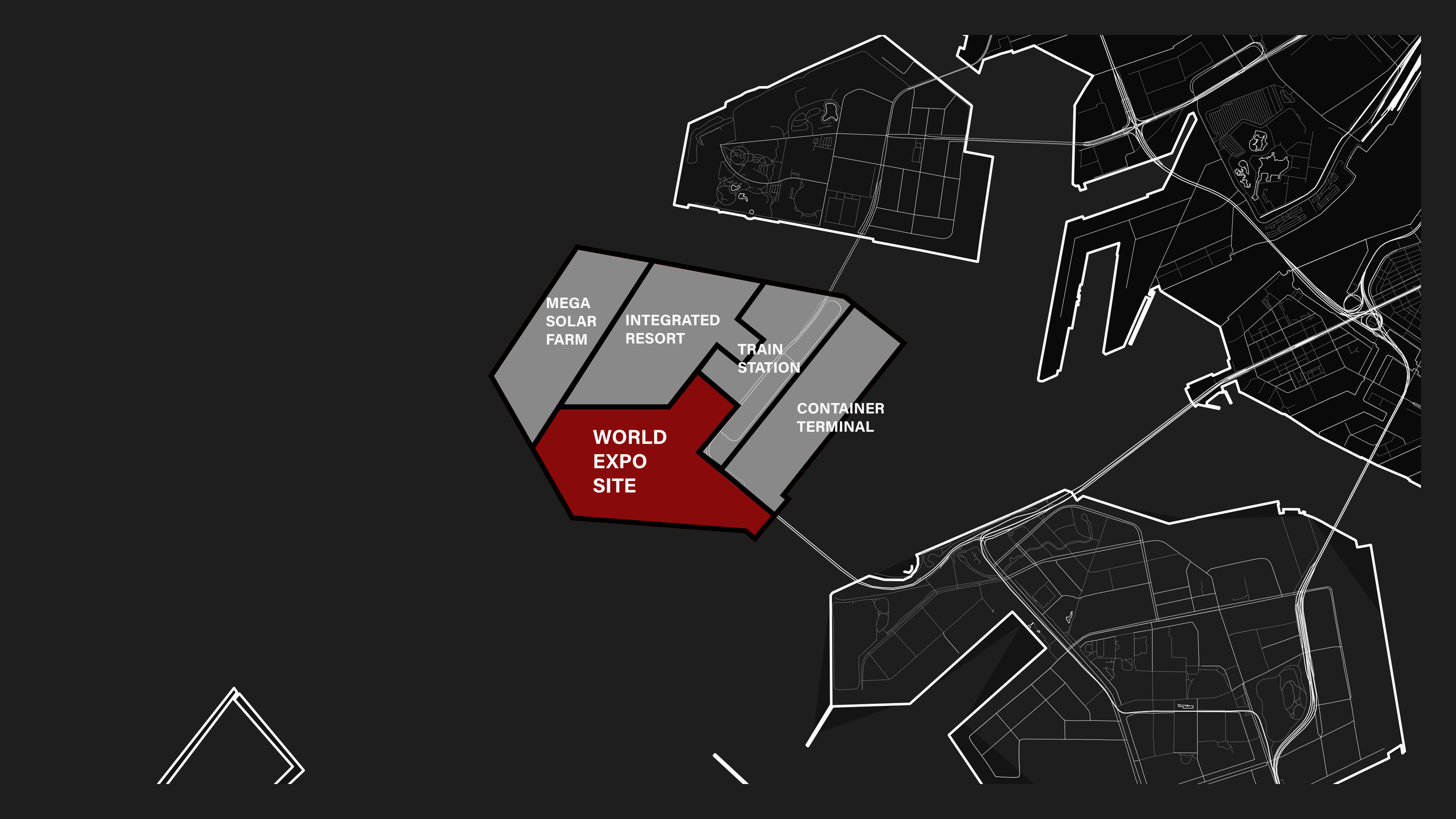

N
E
S
W


S
78.9°
34.8°



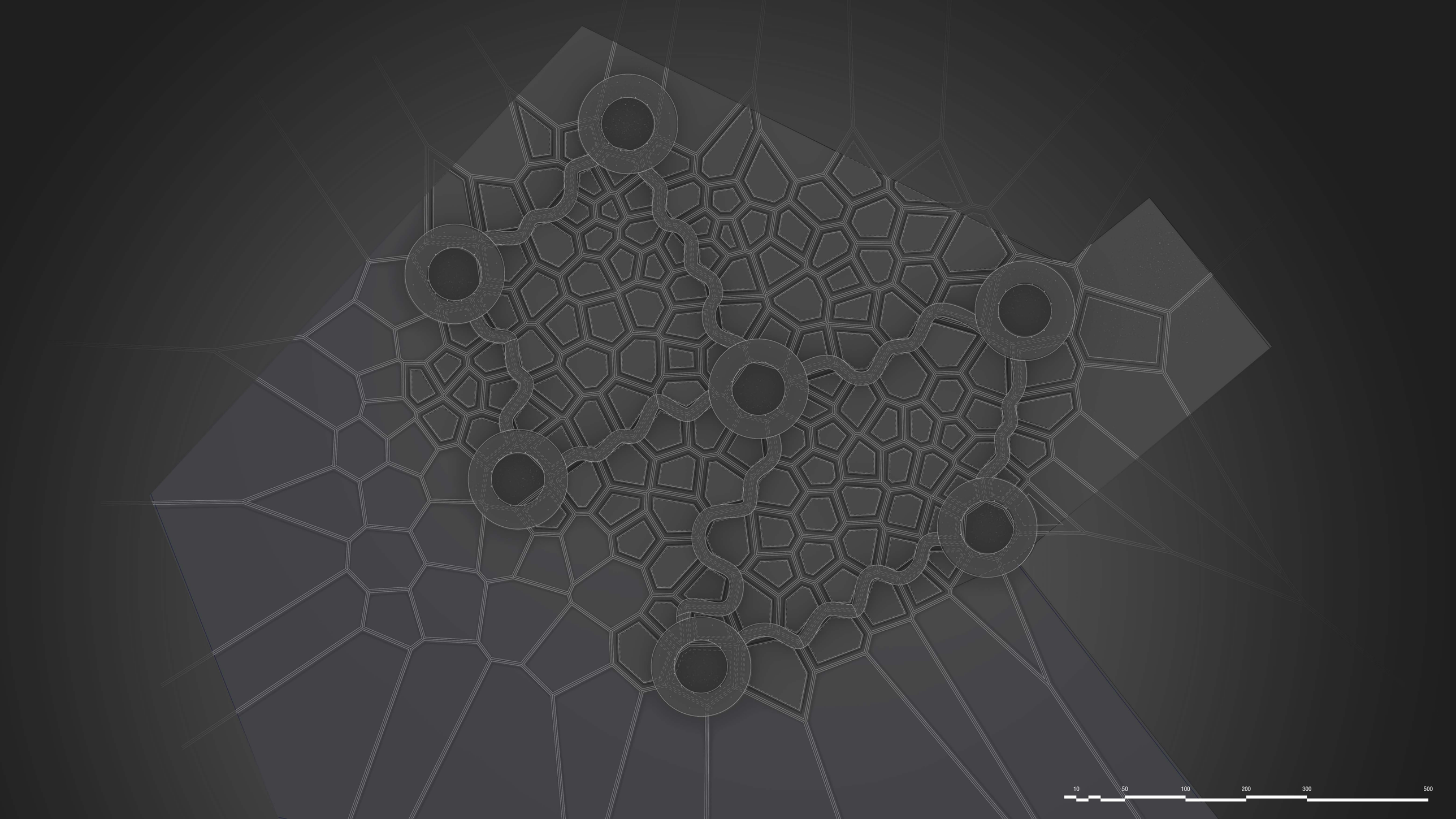

Yumeshima Island

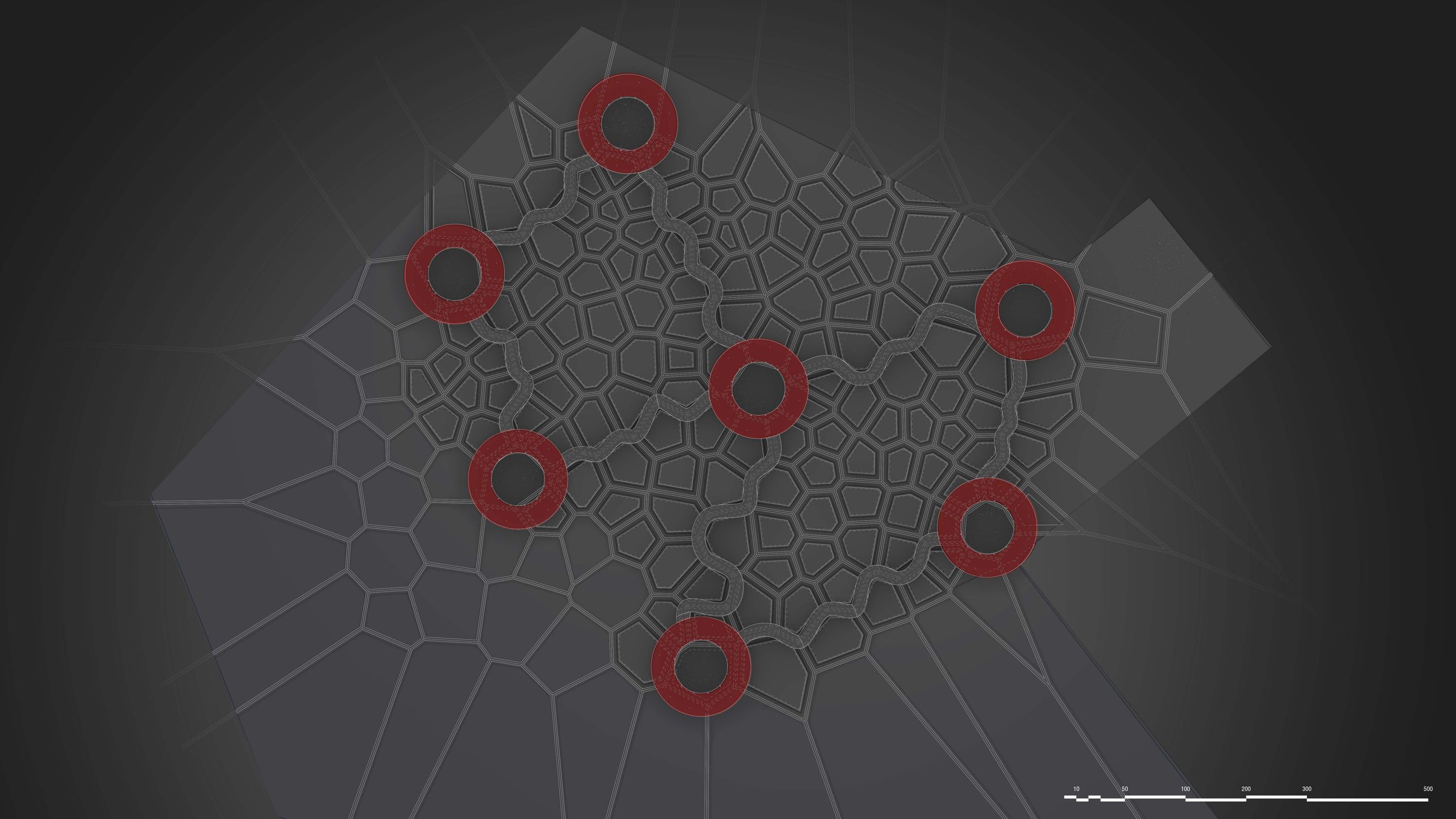



MEGA
SOLAR
FARM
INTEGRATED
RESORT
TRAIN
STATION
CONTAINER
TERMINAL
WORLD
EXPO
SITE


OSAKA-KANSAI
JAPAN
EXPO2025




10
50
100
200
300
500

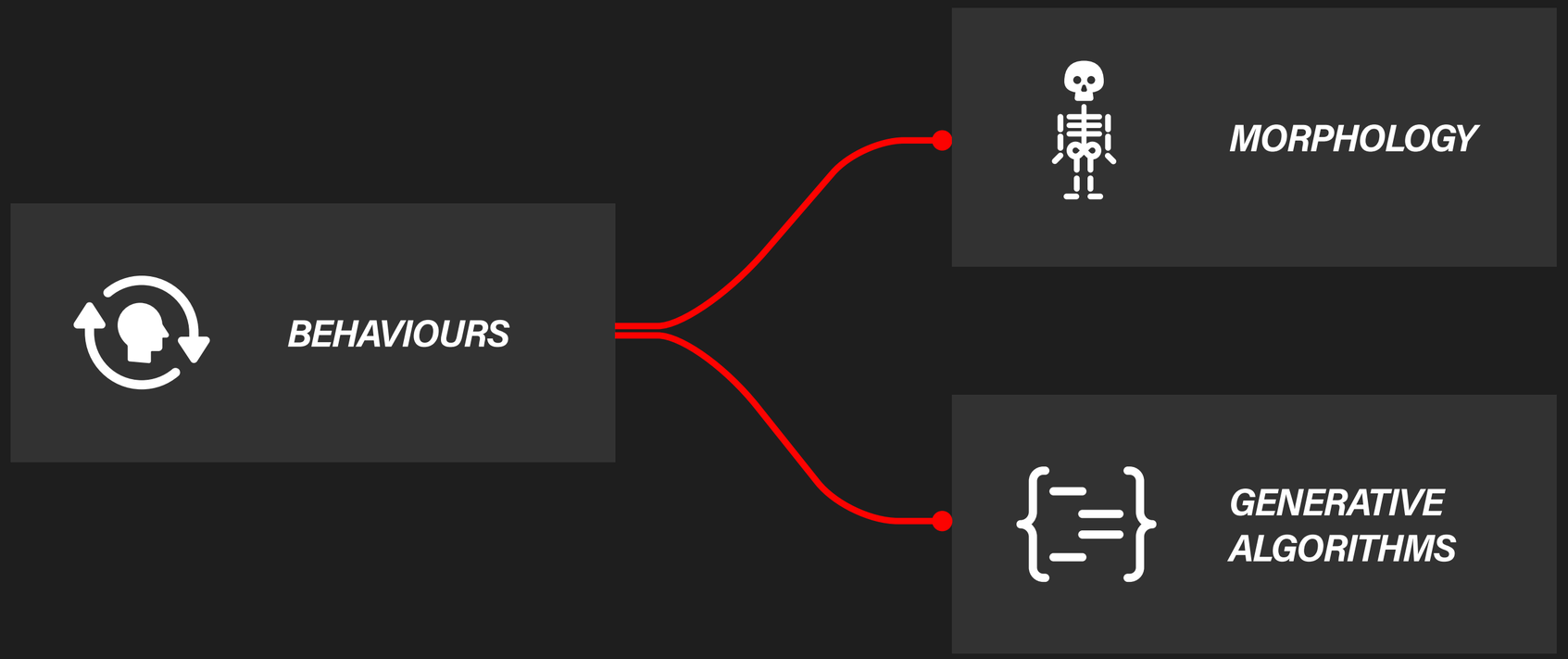

10
50
100
200
300
500



HOW TO DESIGN AN ARCHITECTURE THAT IS ENACTED BY SWARMS OF AUTONOMOUS DRONES?

ARCHITECTURAL INTENTIONS

TRANSLATION

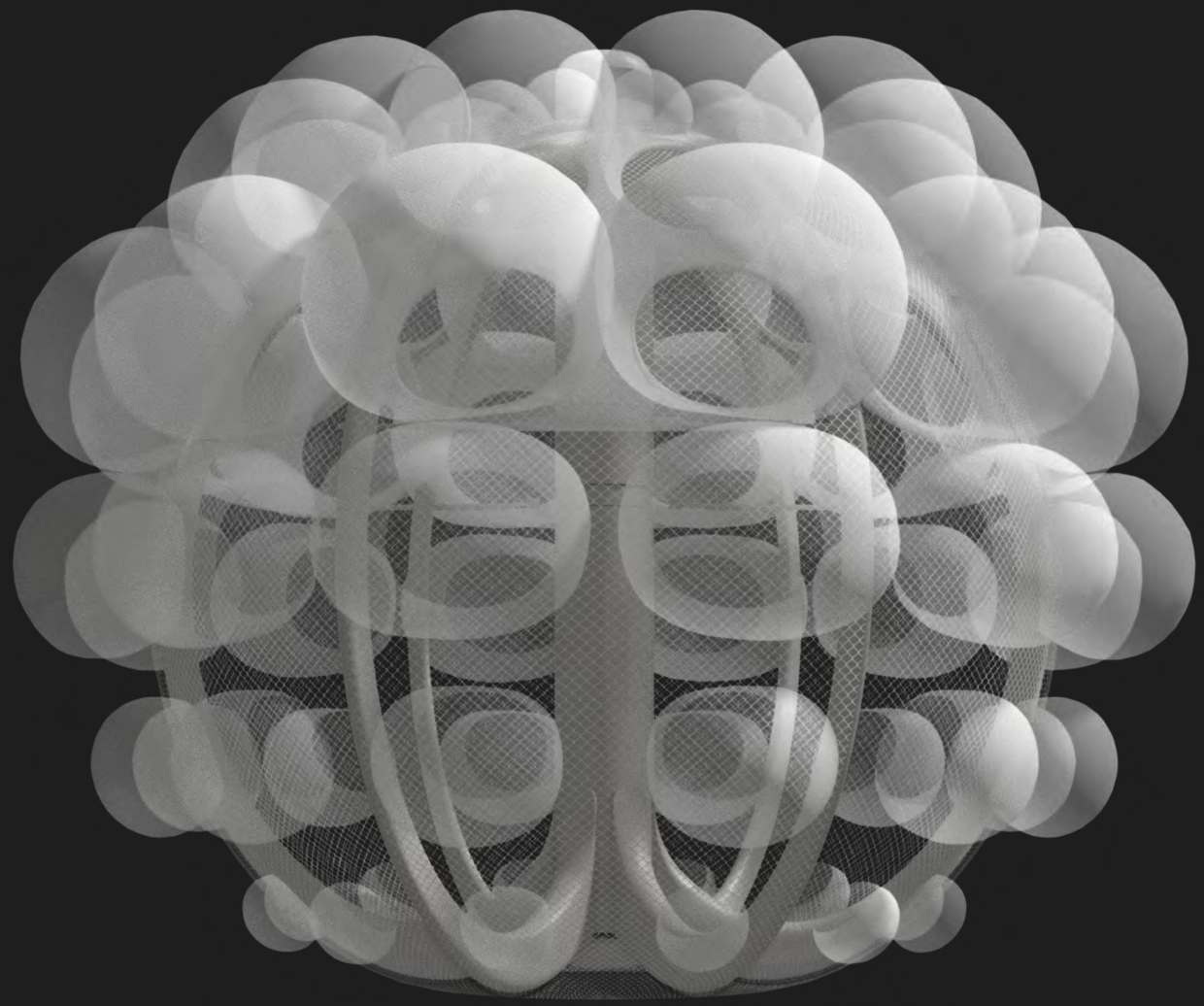





HI!
I'M PIXY

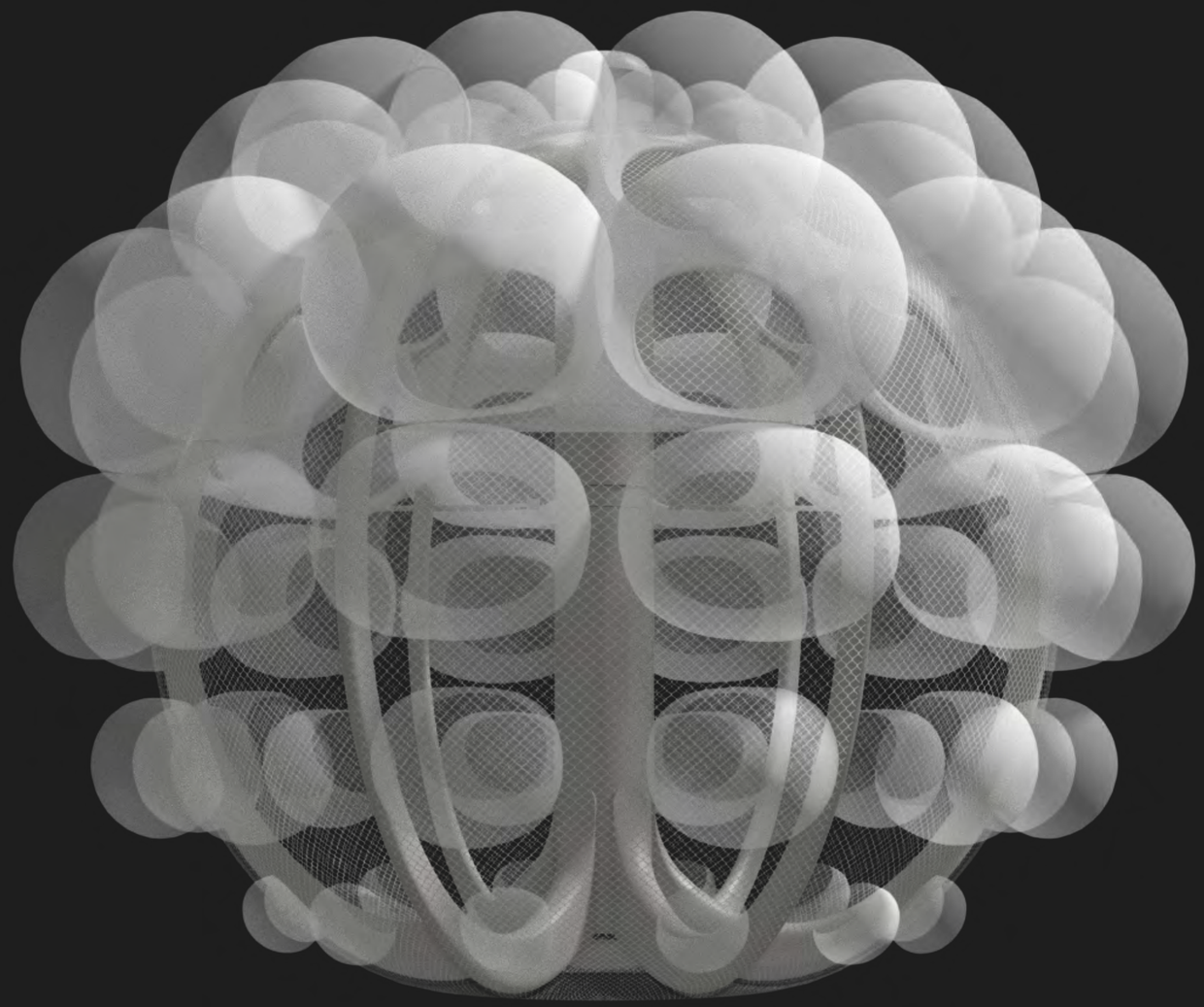
“It’s going to be harder to distinguish: what is alive and what is a machine… And the boundary may start to become meaningless”

Rodney Brooks, “Fast, Cheap & Out of Control”, 1997



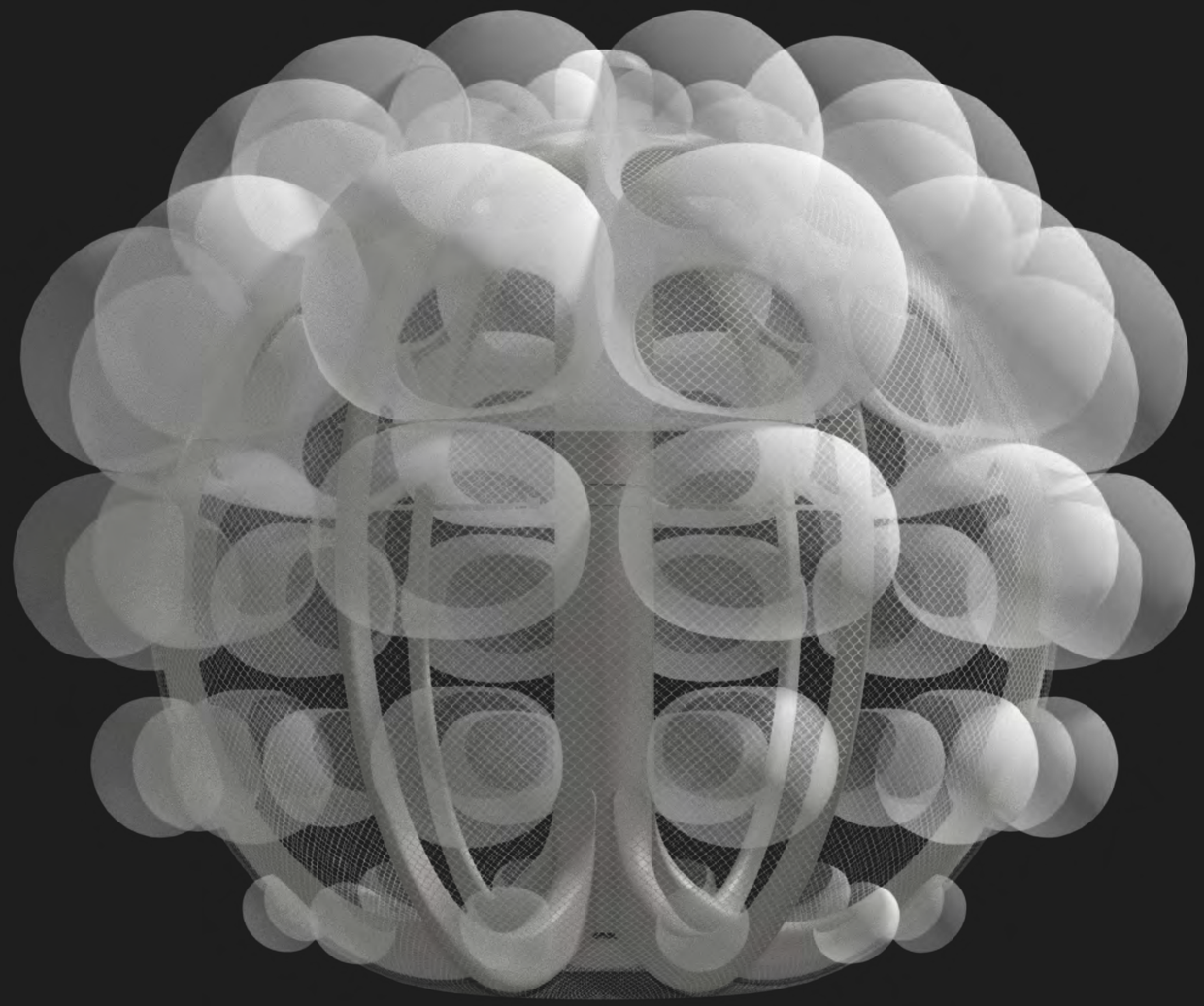


INFLATABLE SKIN



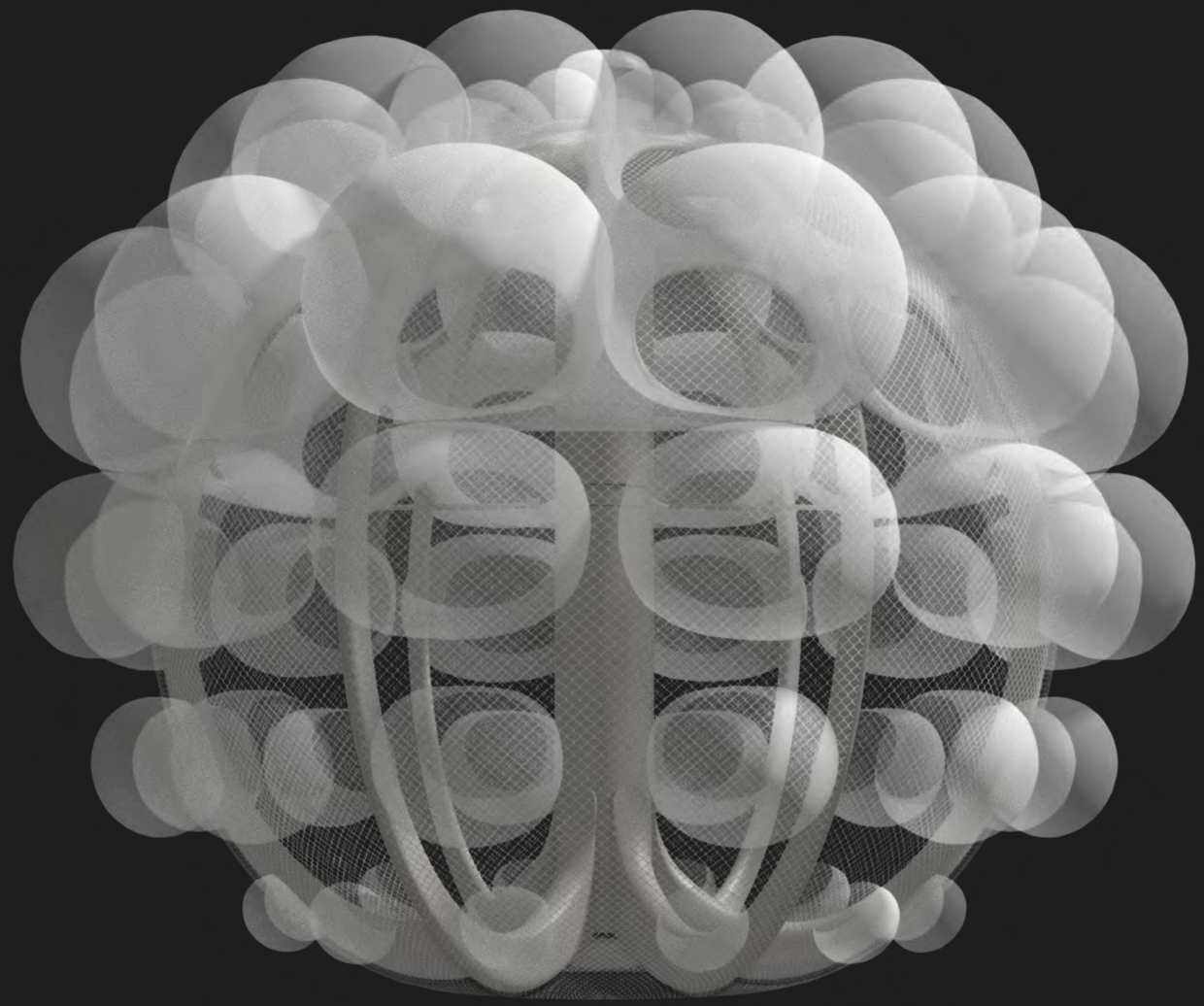


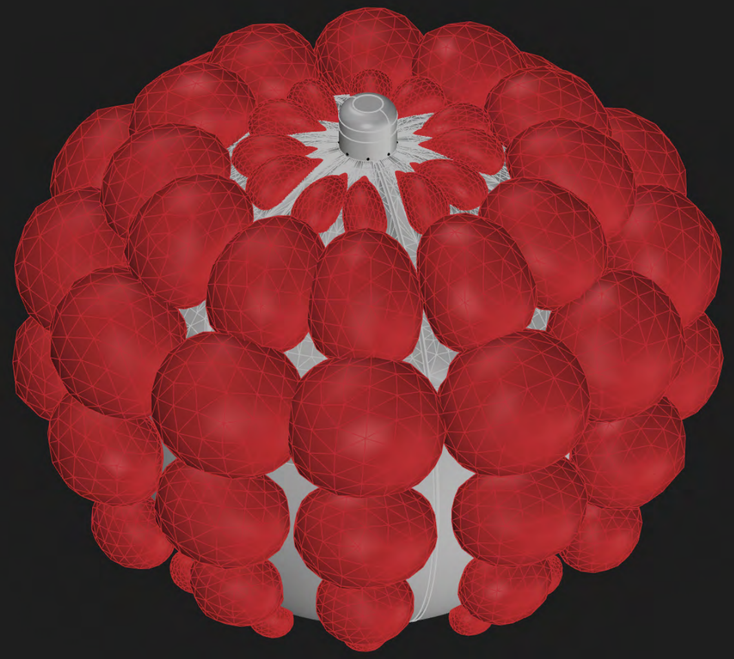
INFLATABLE SKIN - MESH FABRIC



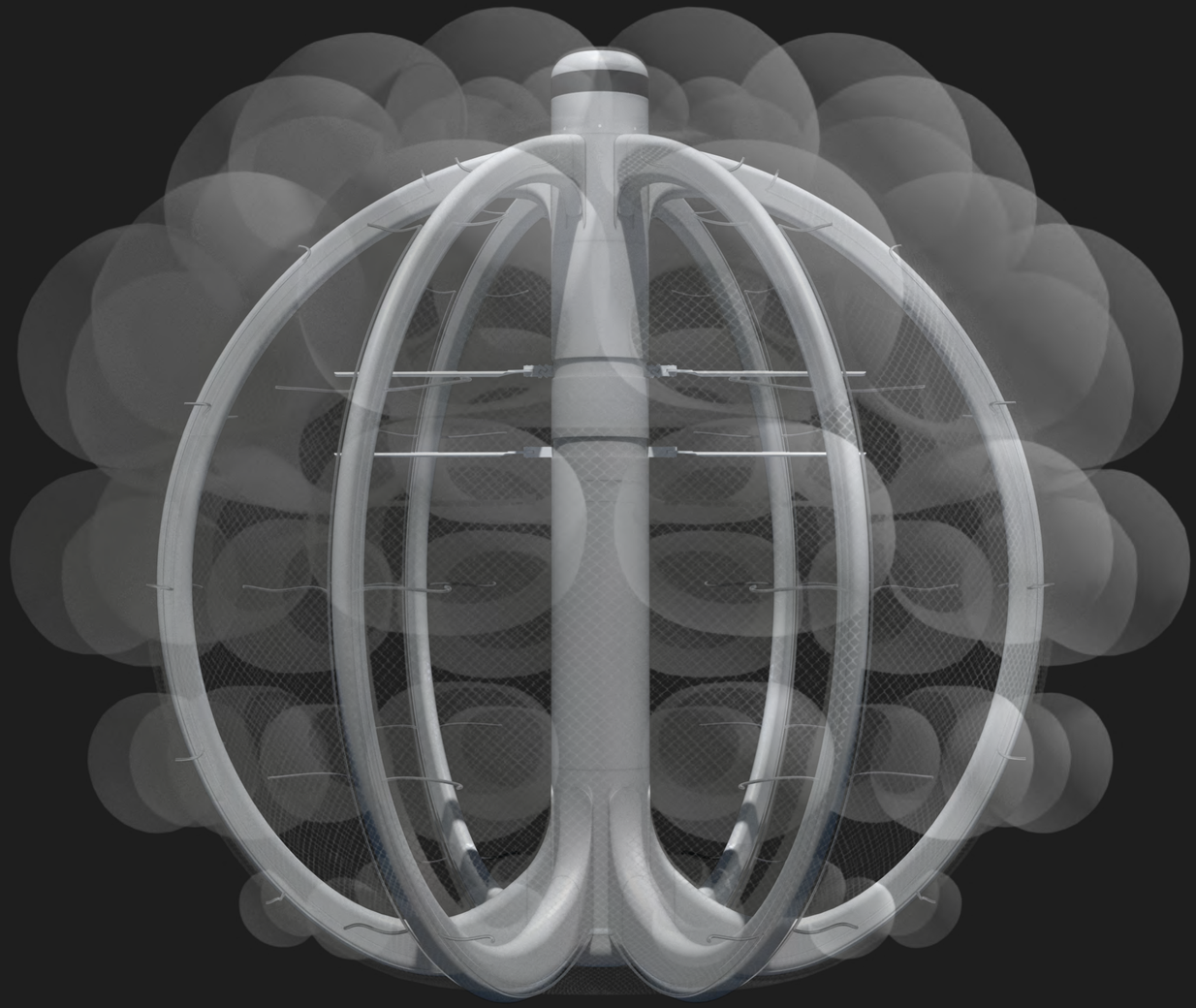


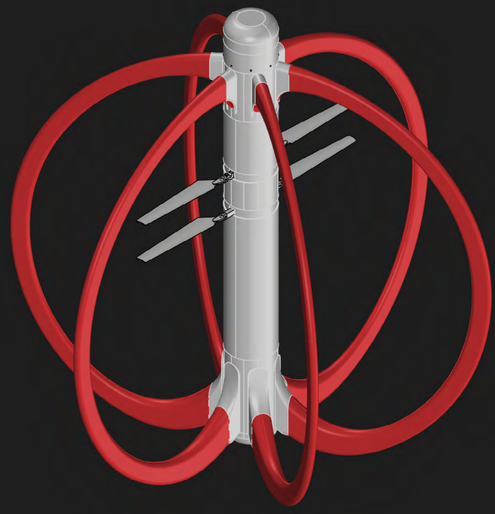
INFLATABLE SKIN - TOP LAMINATION



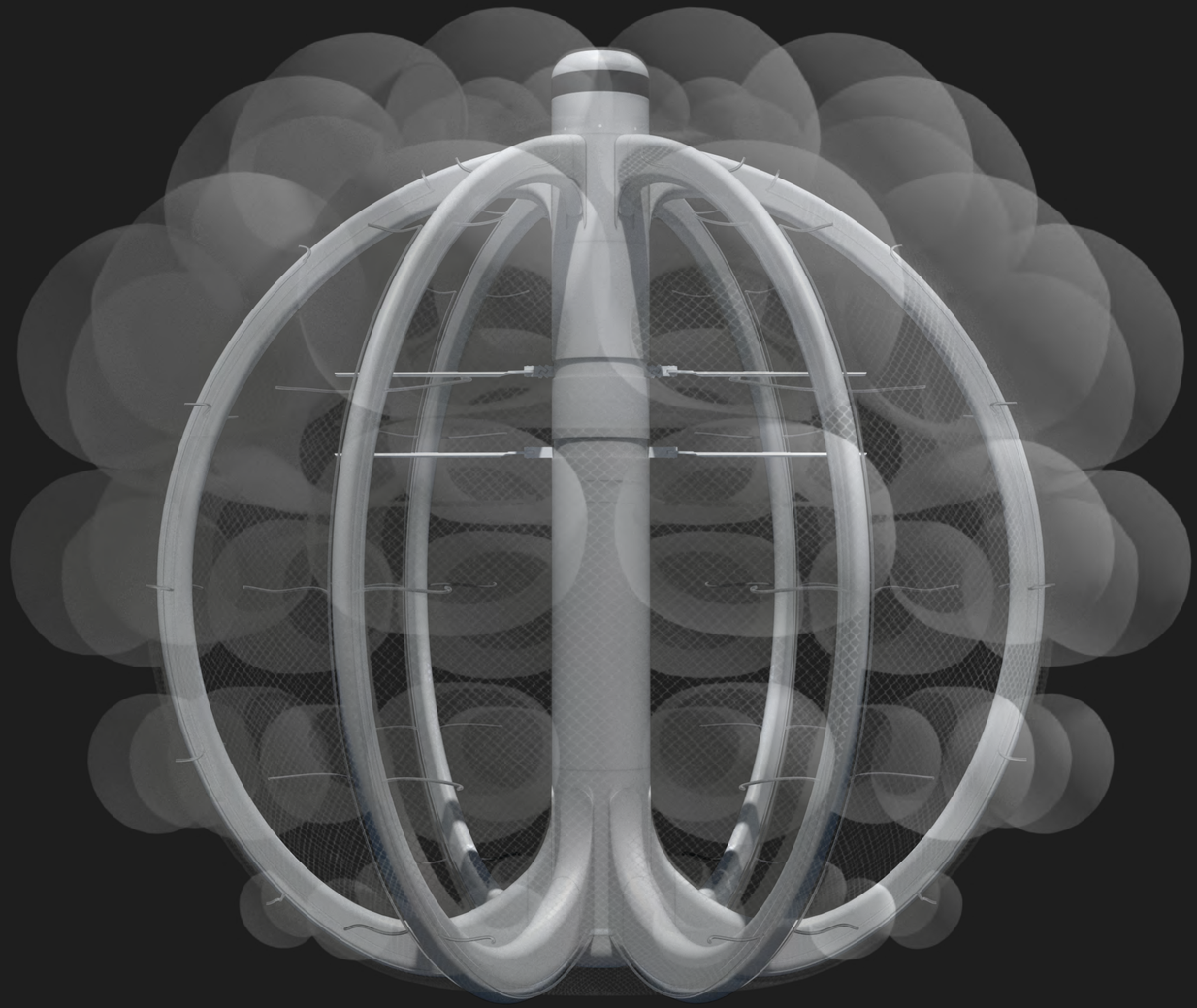


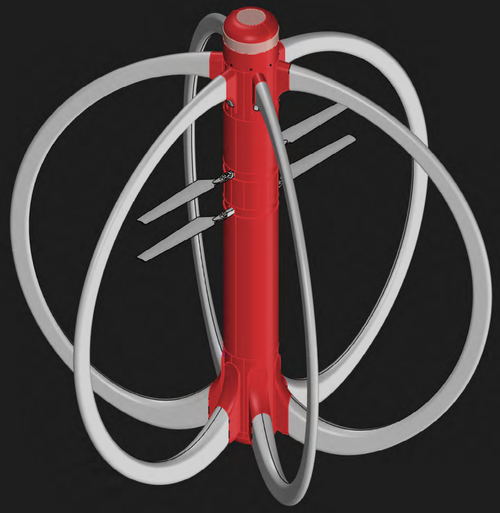
INFLATABLE SKIN



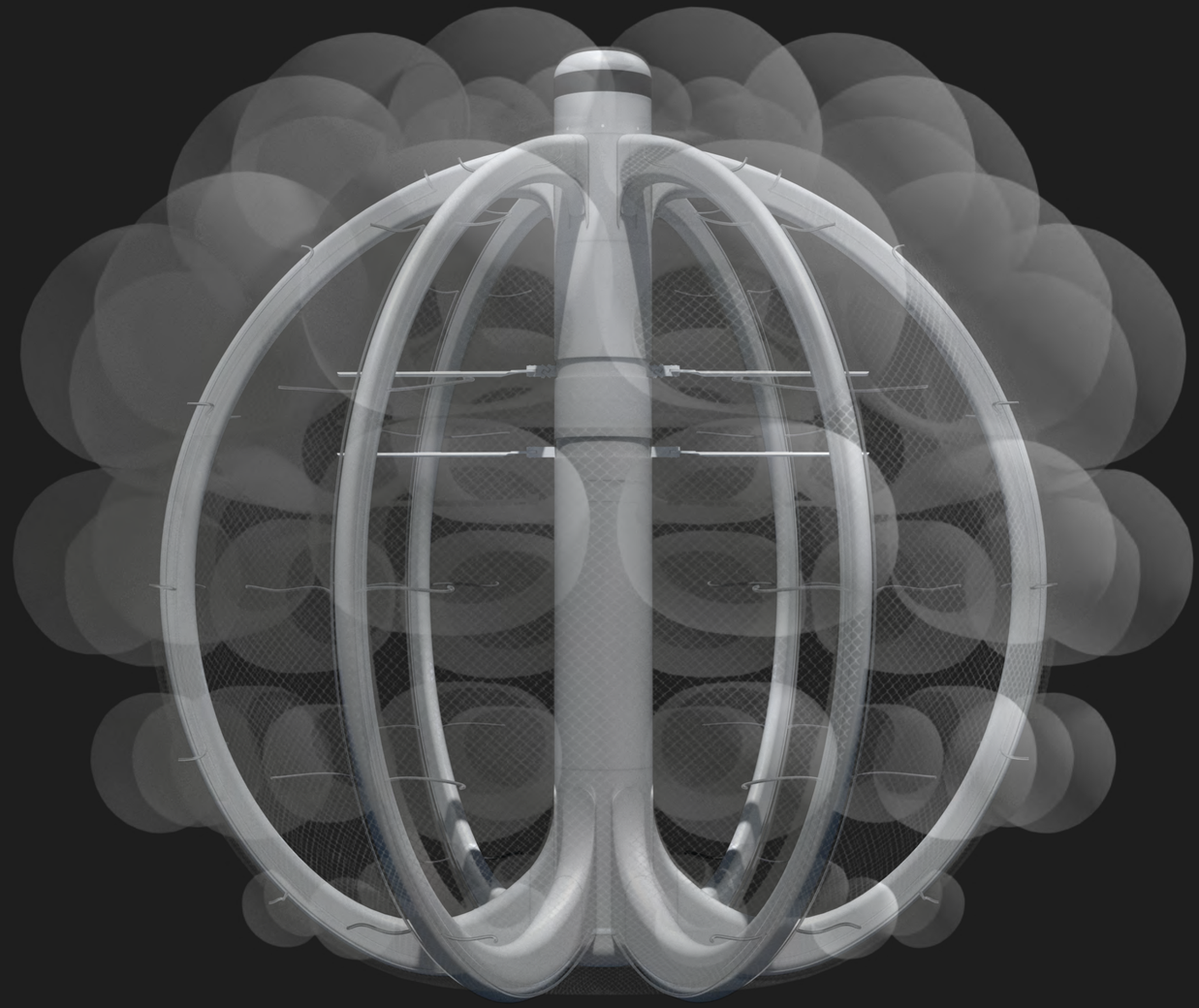


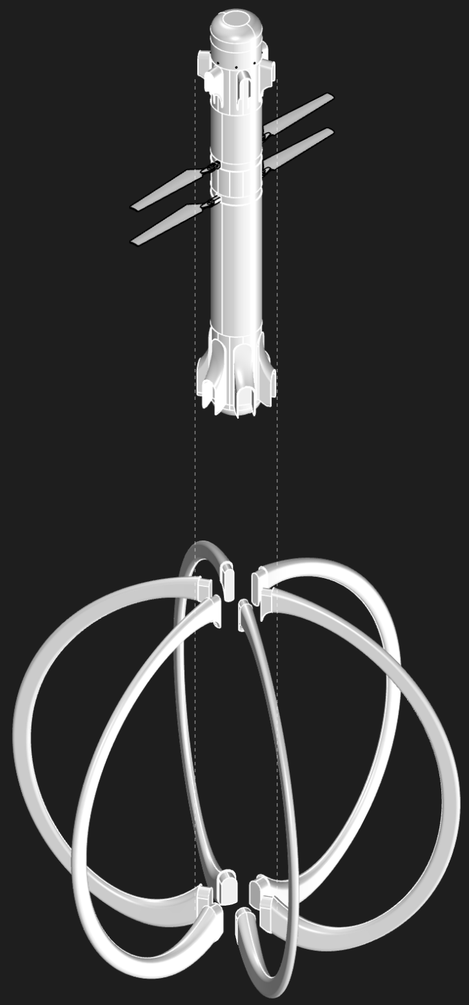
STRUCTURE - FRAME



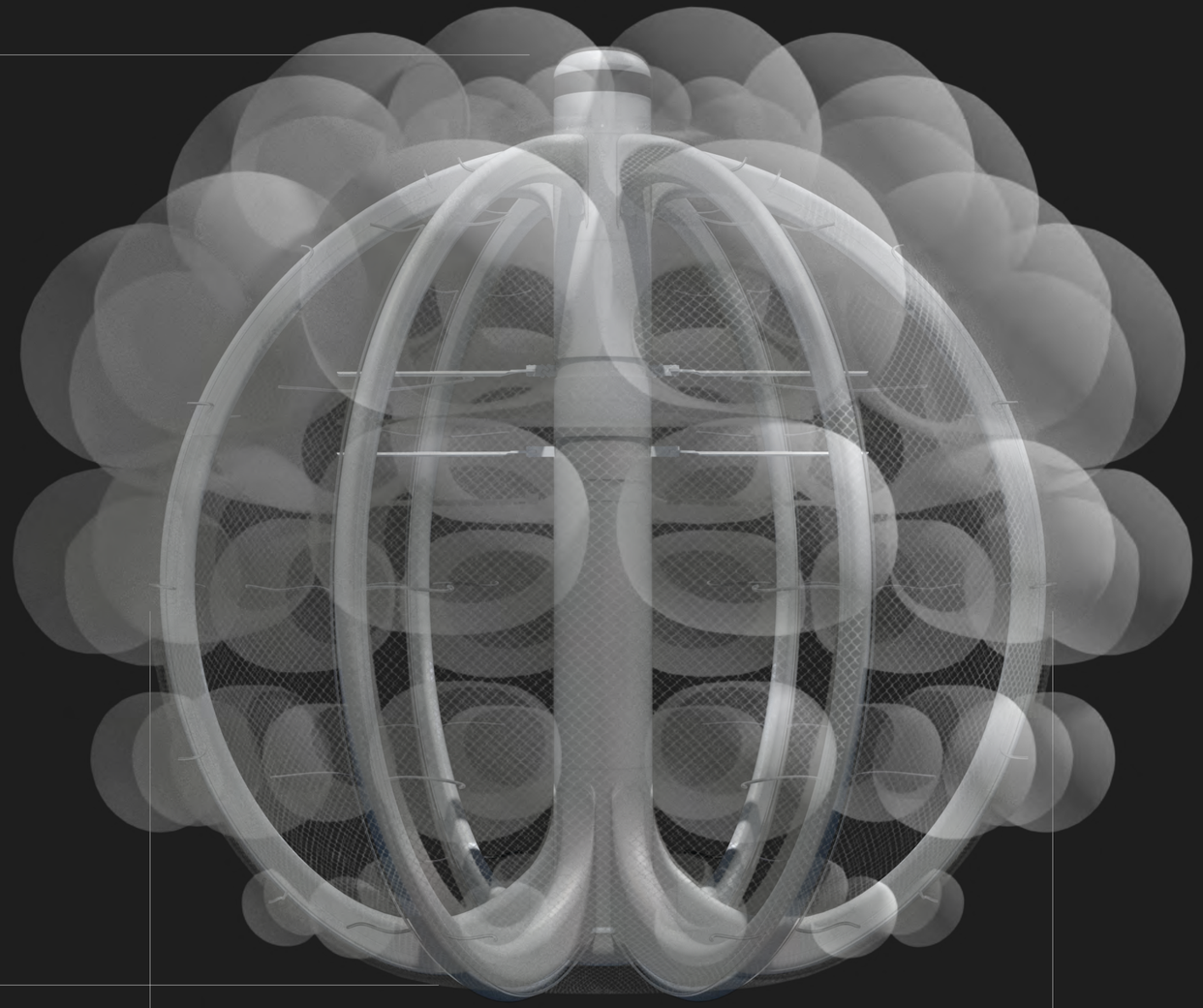


STRUCTURE - CORE





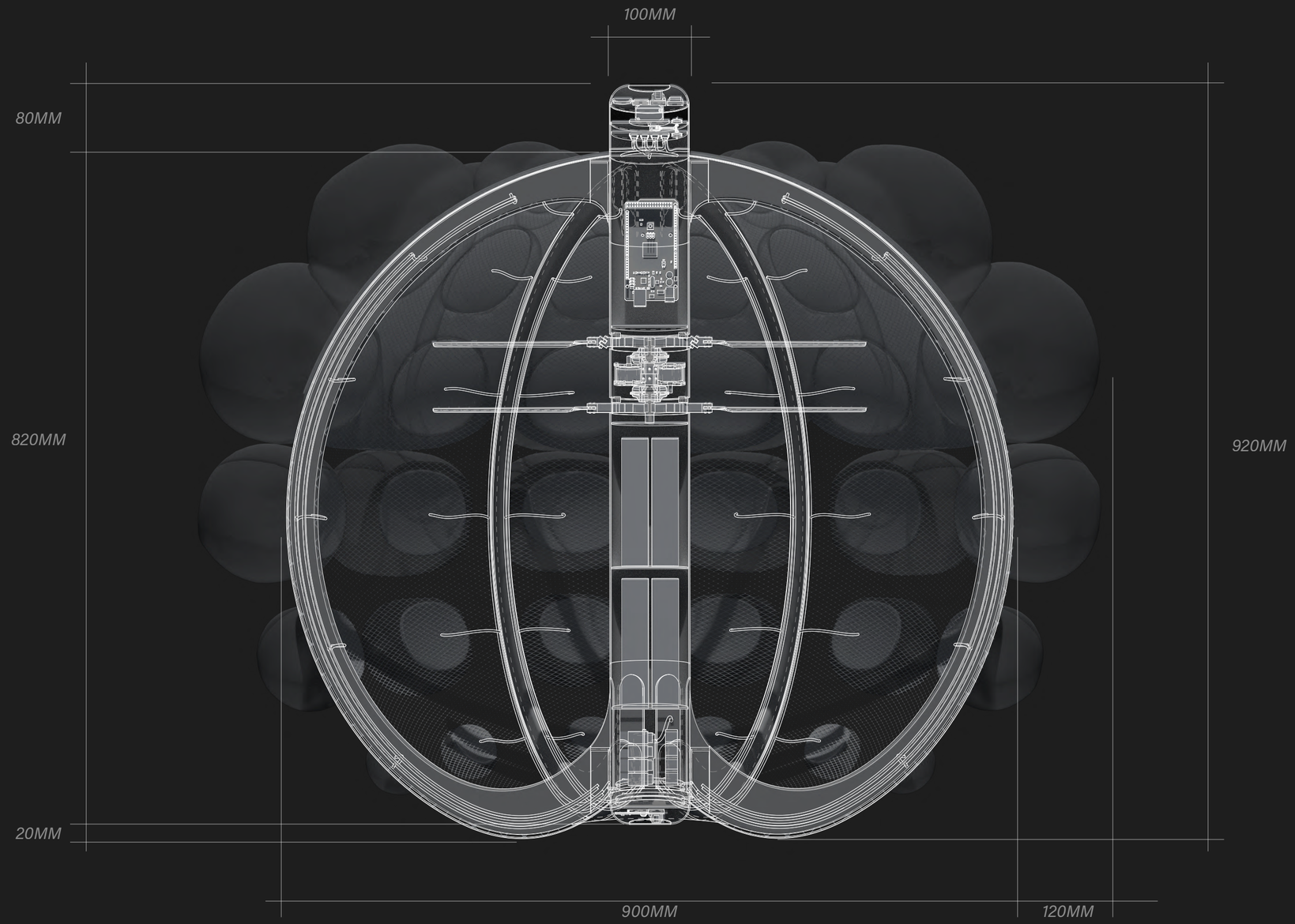
STRUCTURE - SLIDE CONNECTION



WEIGHT 3KG

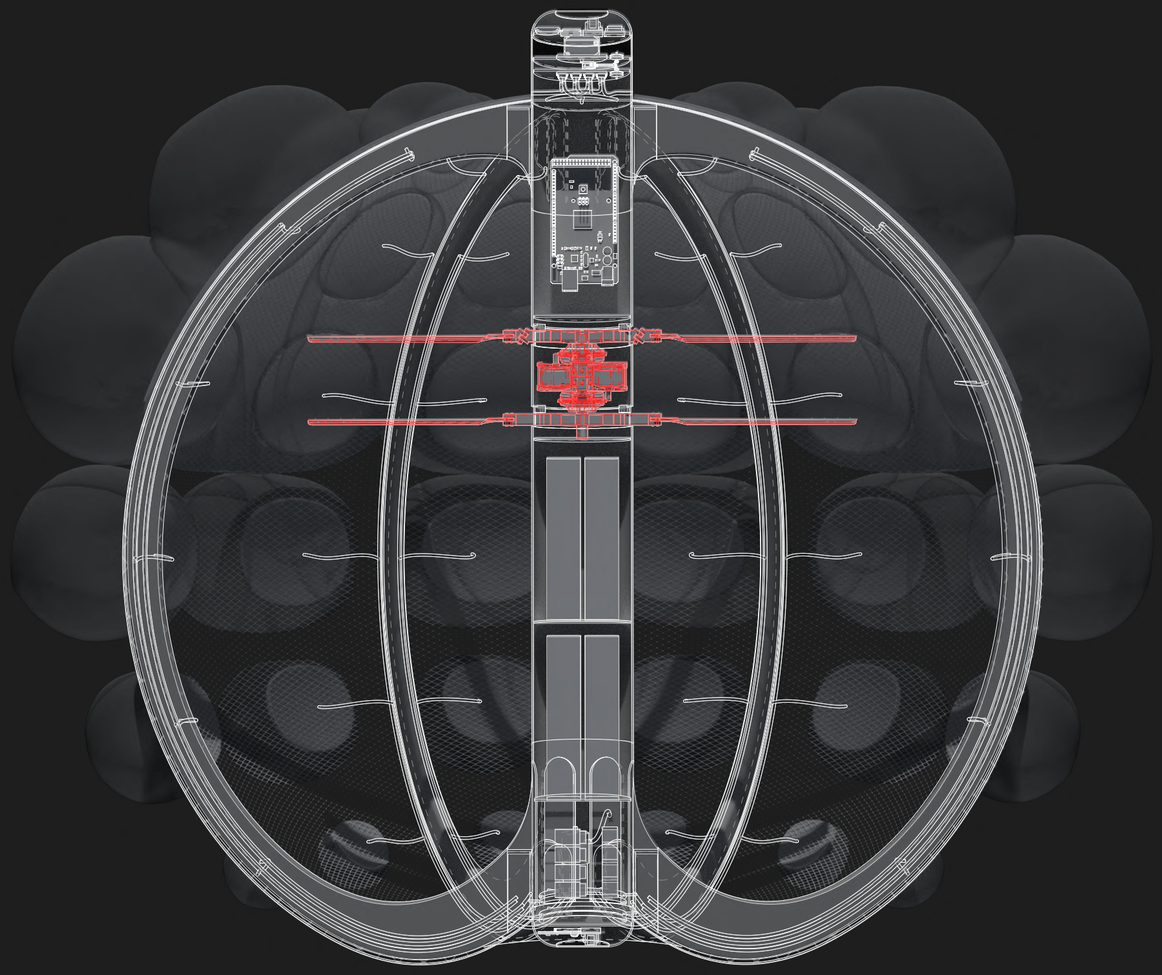


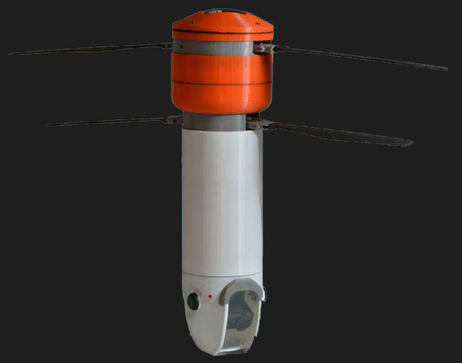
WEIGHT 1.4KG



WEIGHT	
INFLATABLE SKIN	***1,000g***
FRAME+ CORE	***320g***
PROCESSING	***45g***
BATTERY	***900g***
VISION	***170g***
ENVIRONMENT	***25g***
AVIONICS	***250g***
PNEUMATIC	***220g***
LIGHTING	***70g***

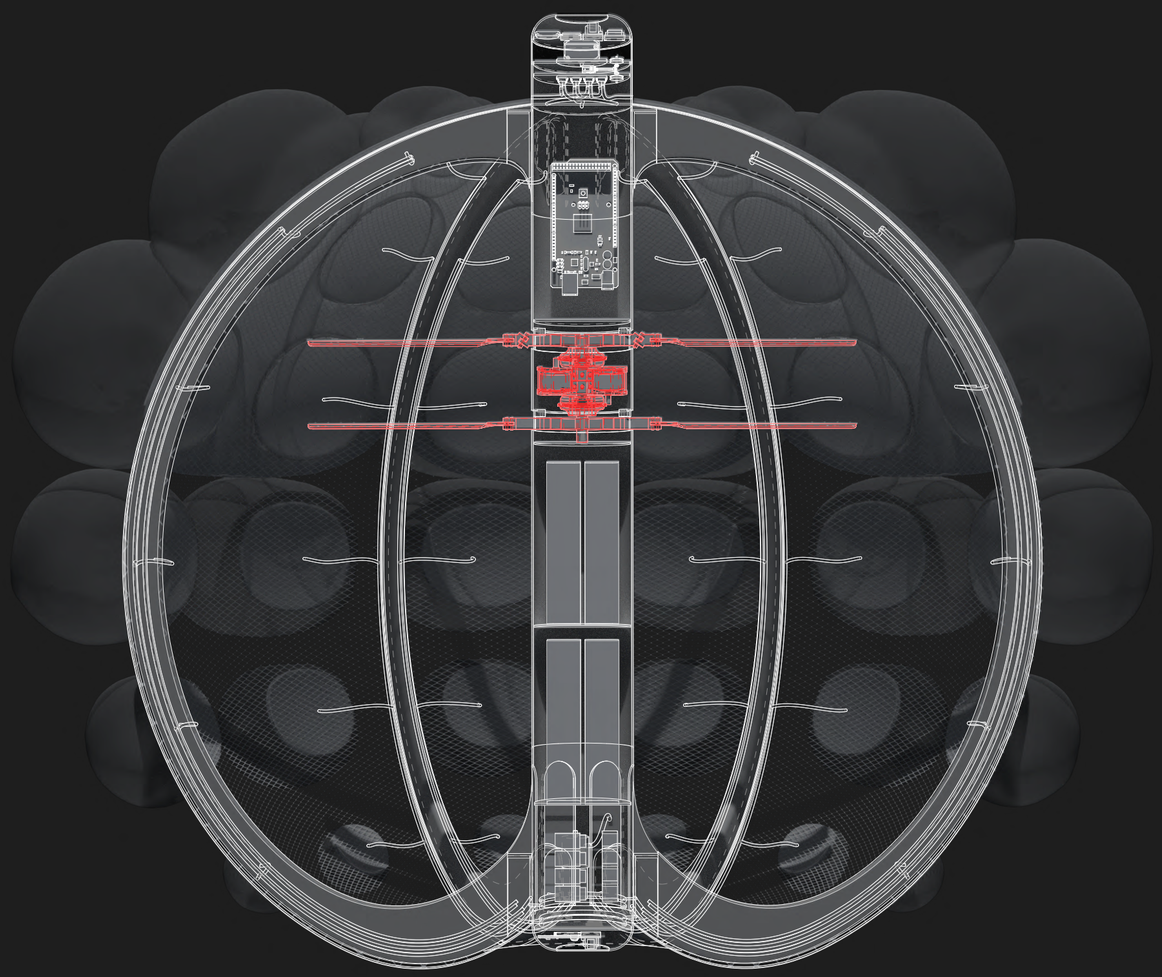
TOTAL:	***3,000 grams***

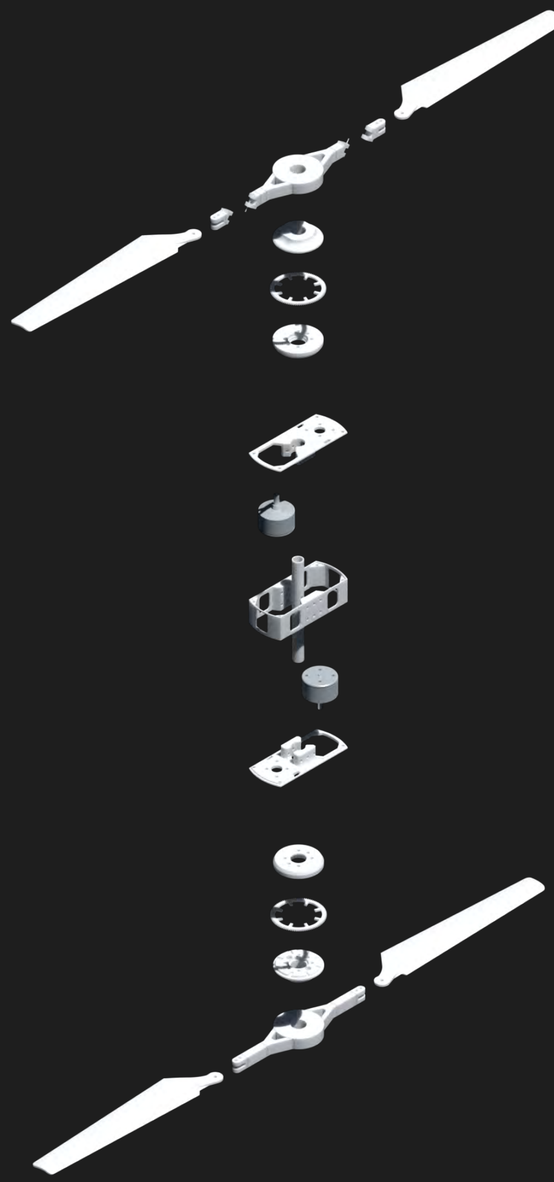


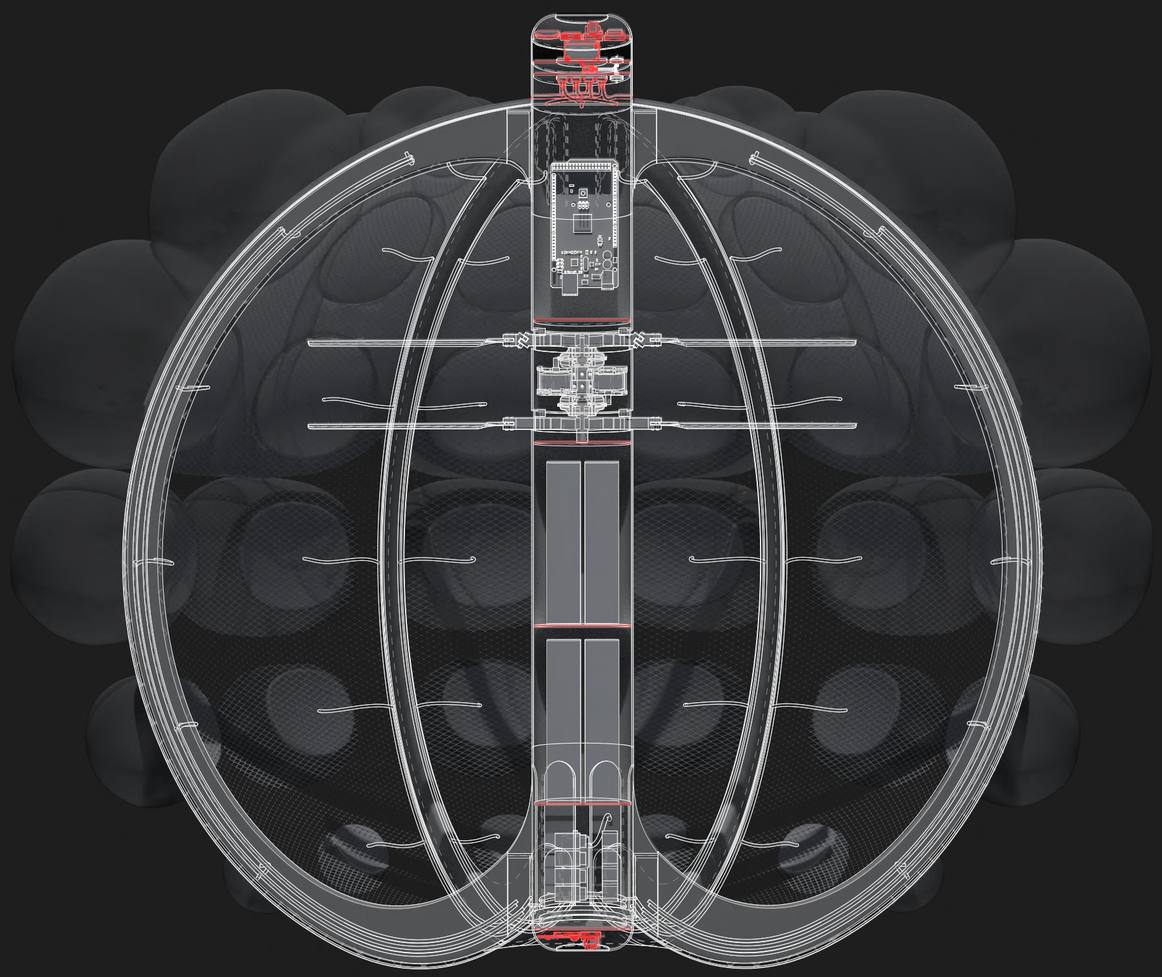




TOP: T-DRONE, OPEN-SOURCE COAXIAL UAV
BOTTOM: SPIRIT DRONE, ASCENT AEROSYSTEMS





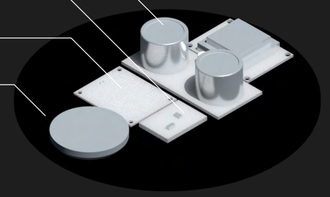


ULTRASONIC SENSOR

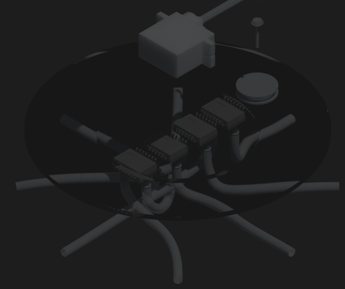
DAYLIGHT SENSOR

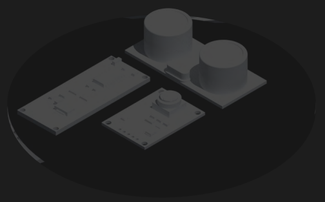
RAIN SENSOR

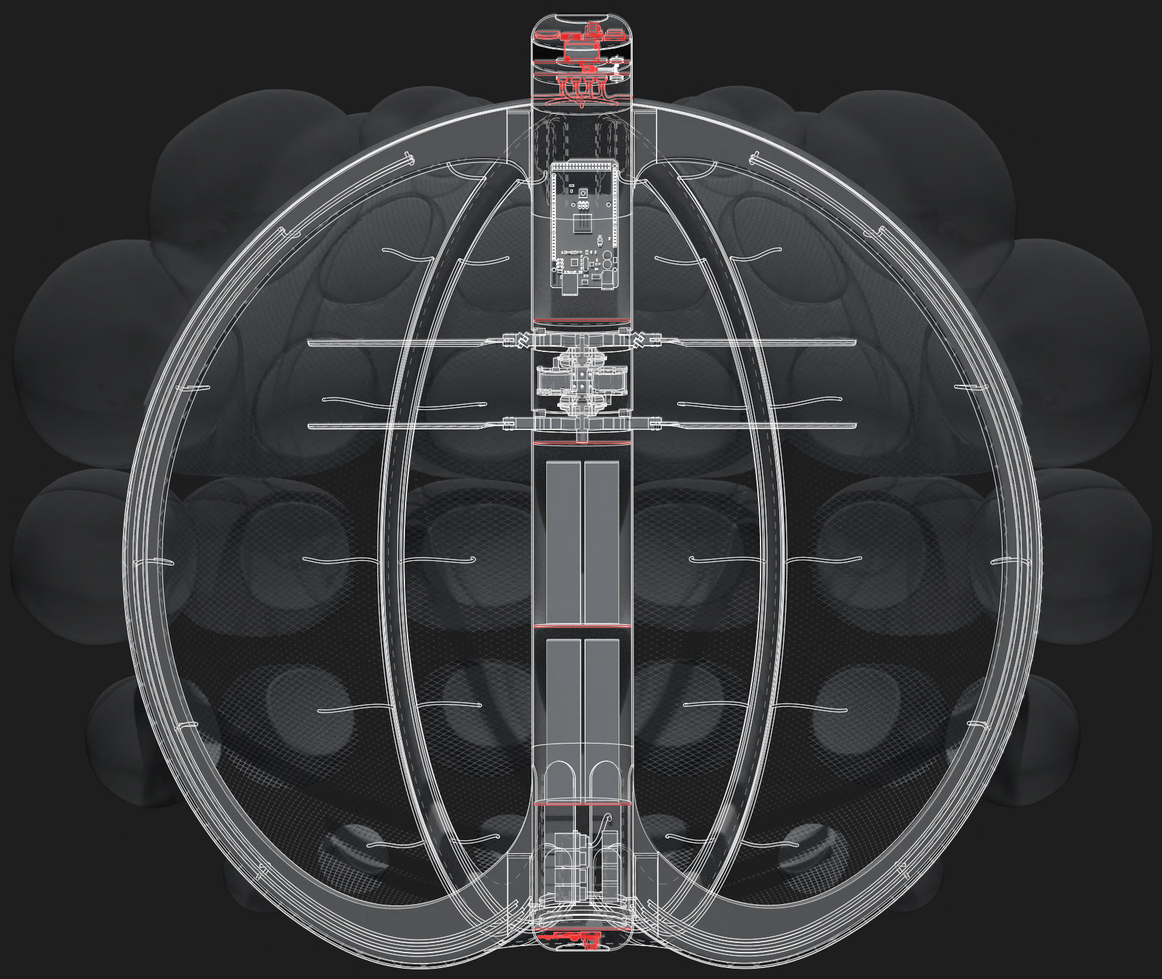
GPS

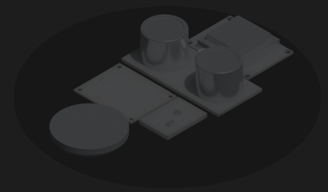


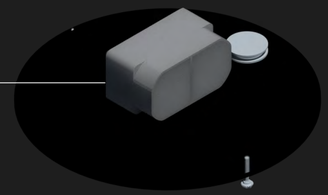




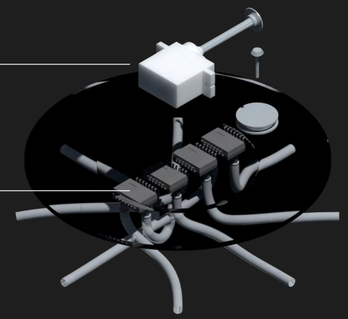






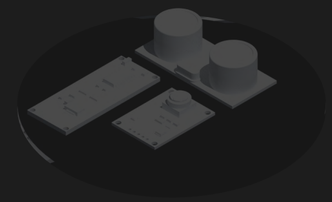


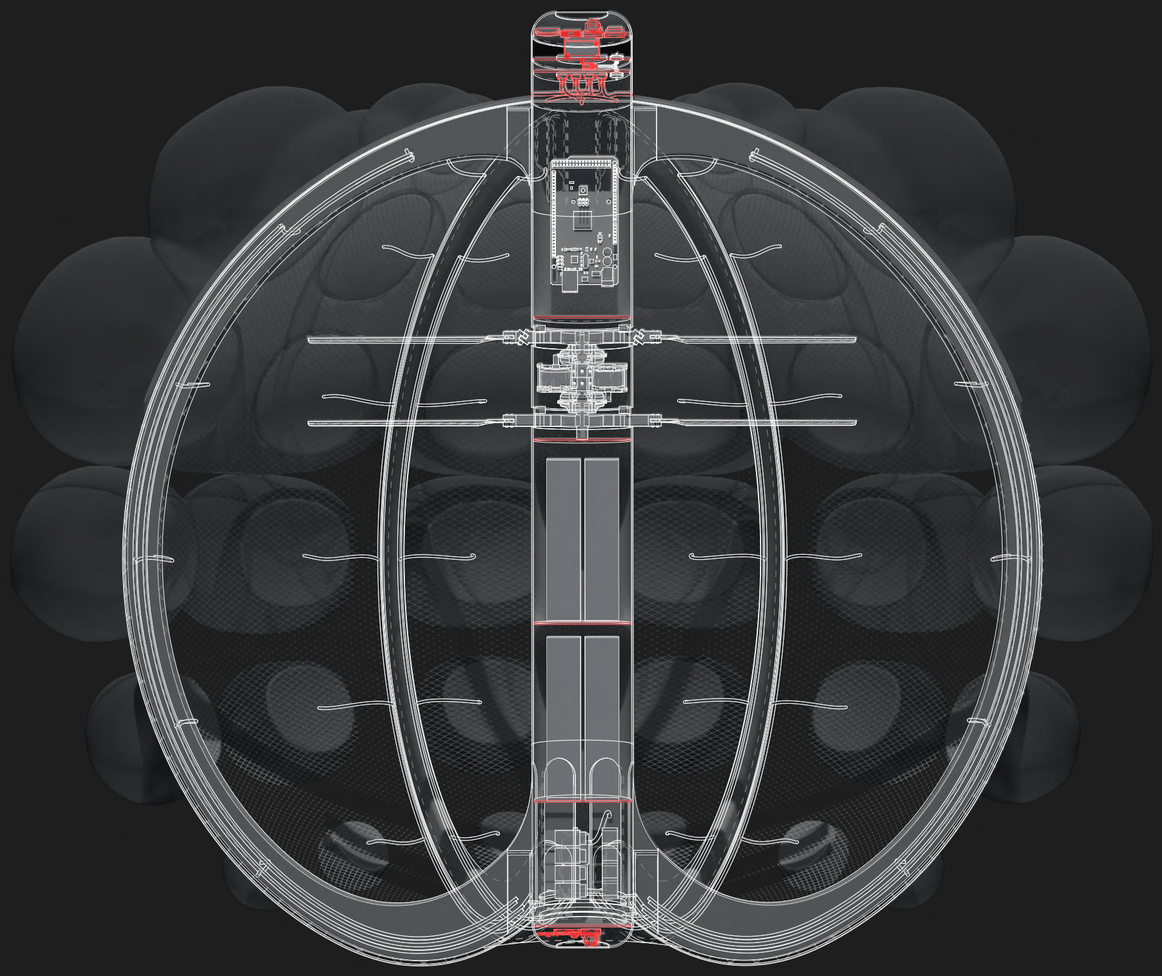
LIDAR

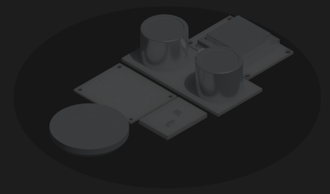


SERVO MOTOR

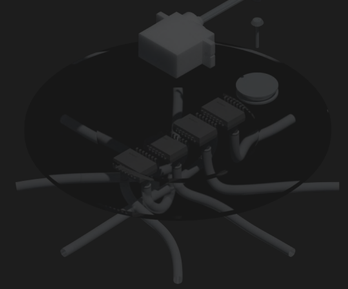
ANEMOMETER







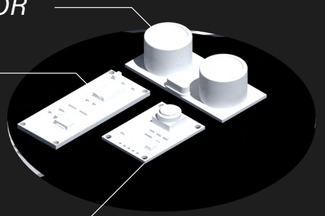


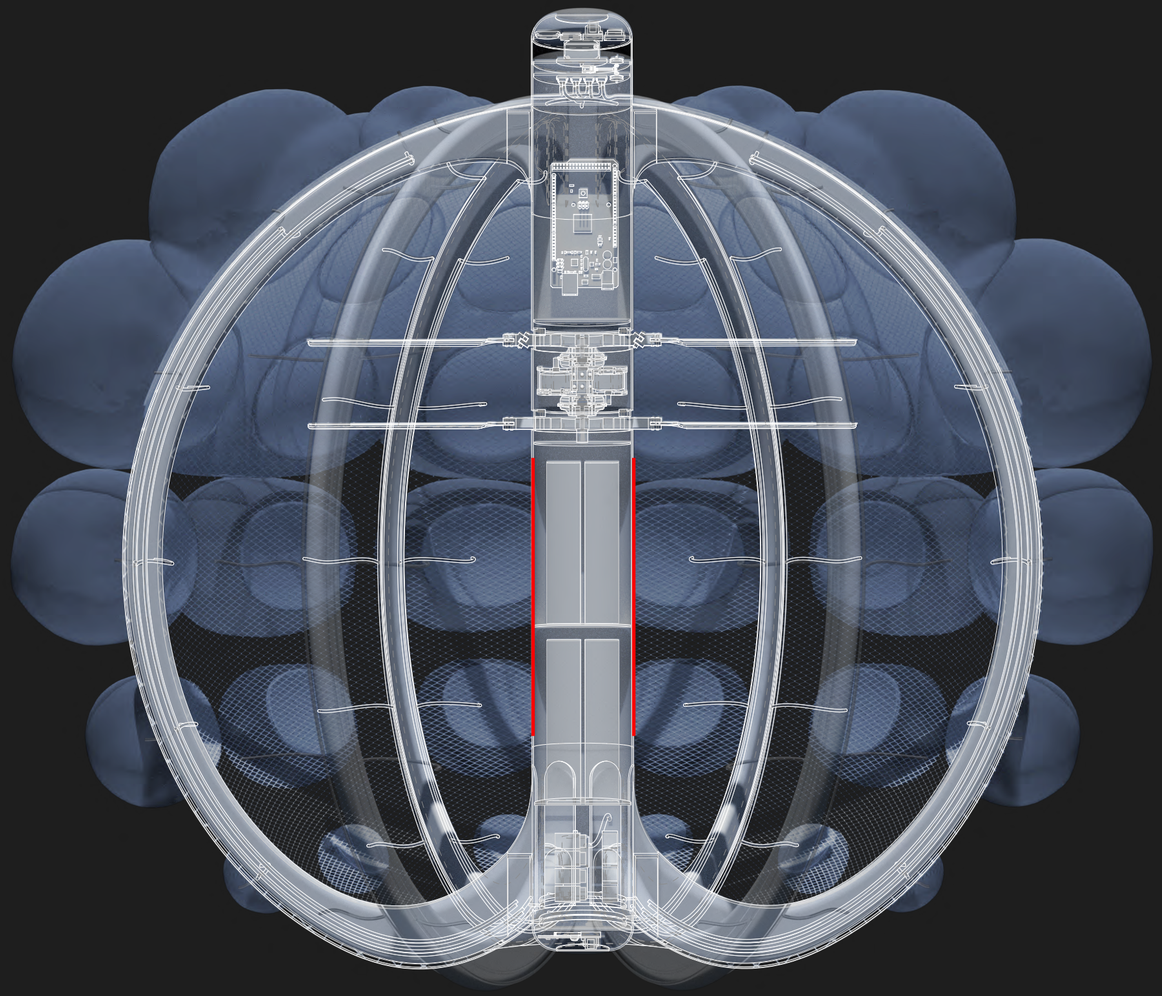


ULTRASONIC SENSOR

COMMUNCATIONS
+AMBIENT LIGHT
+MICROPHONE

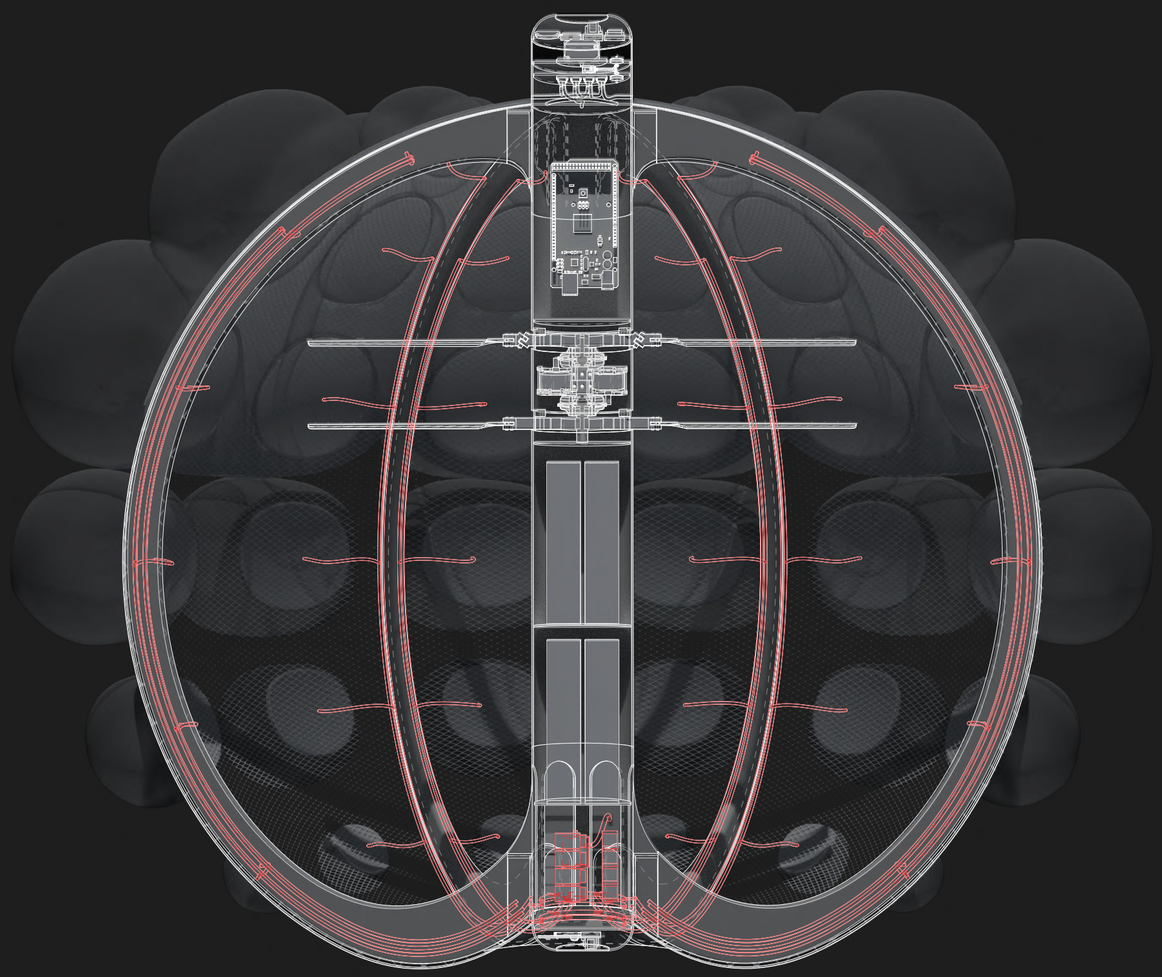
CAMERA

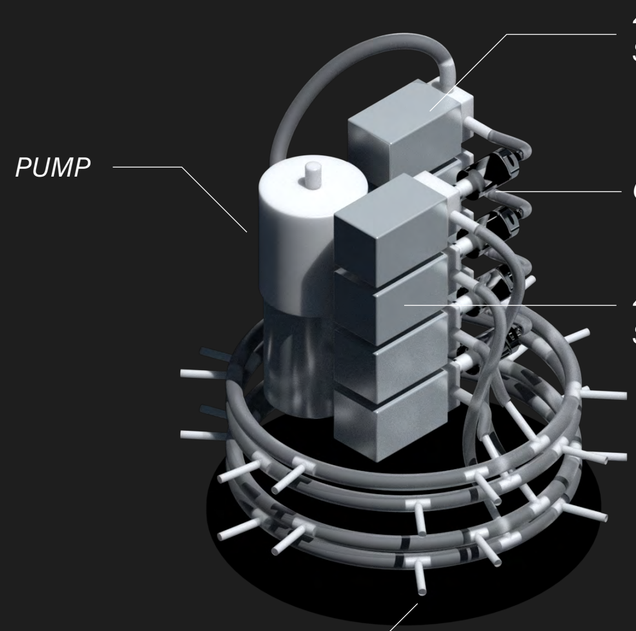


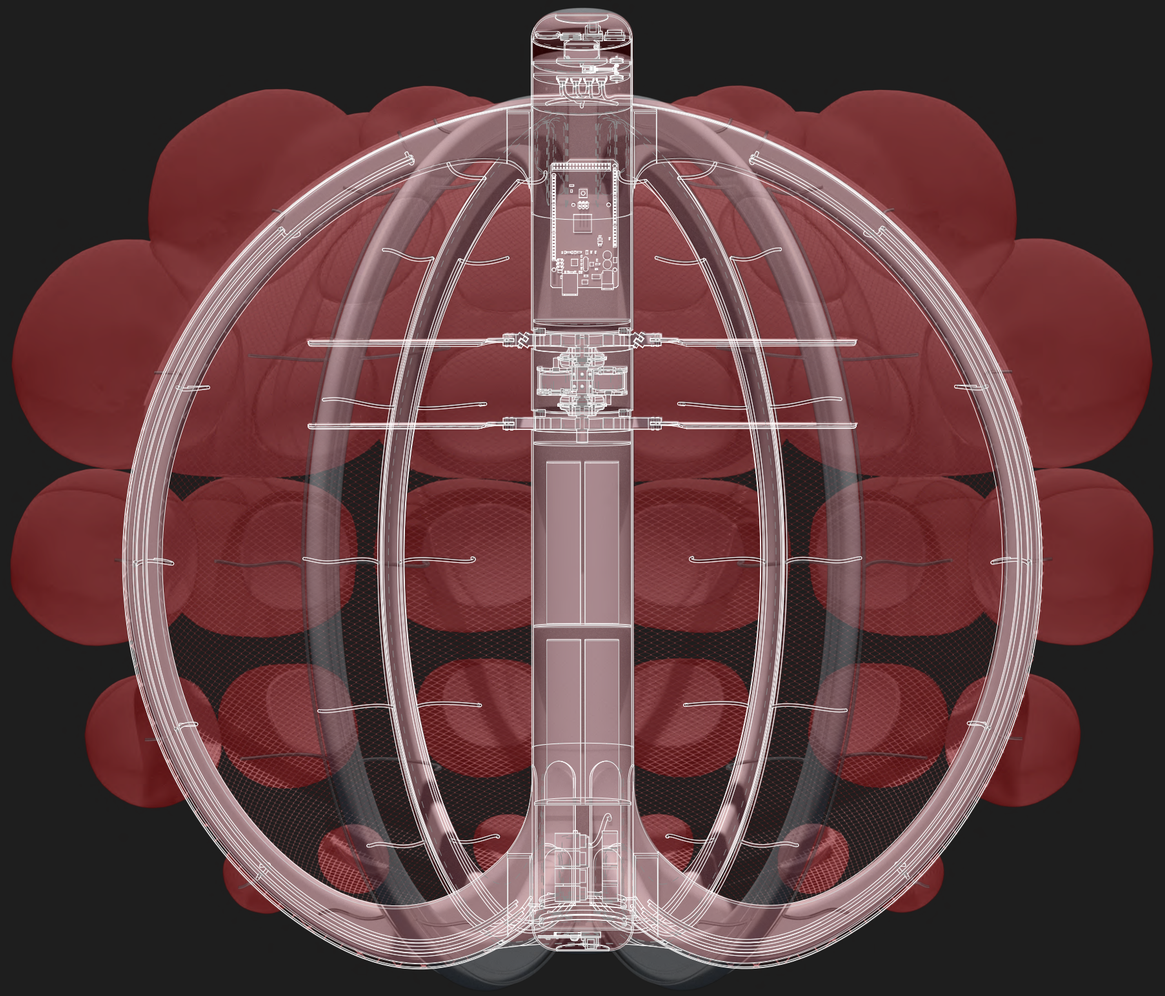


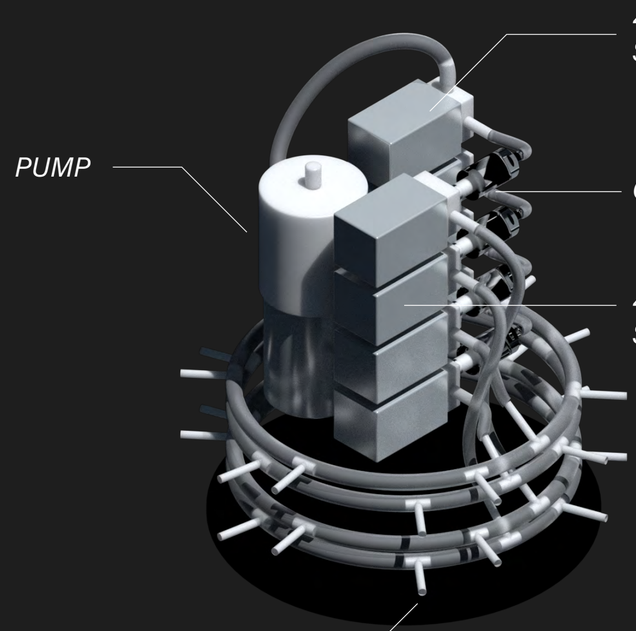
LIGHTING
6X RGB LED STRIPS, LENGTH 600MM

SELF-EXPRESSION

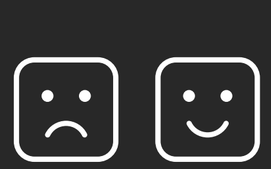



4-WAY
SOLENOID VALVE
PUMP
ONE WAY-VALVE
4 SINGLE
SOLENOID VALVES
TUBING
+ CONNECTORS

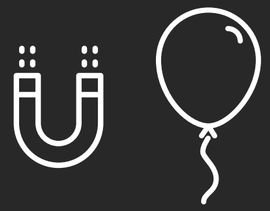



4-WAY
SOLENOID VALVE
PUMP
ONE WAY-VALVE
4 SINGLE
SOLENOID VALVES
TUBING
+ CONNECTORS

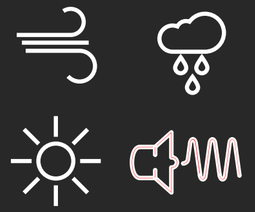
INFLATABLE SKIN



CREATURE'S SELF-EXPRESSION



CONNECTION WITH OTHER CREATURES

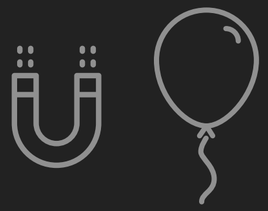


ENVIRONMENT MODULATION

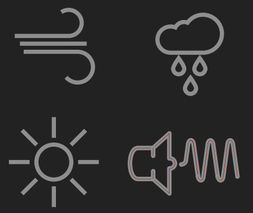
INFLATABLE SKIN



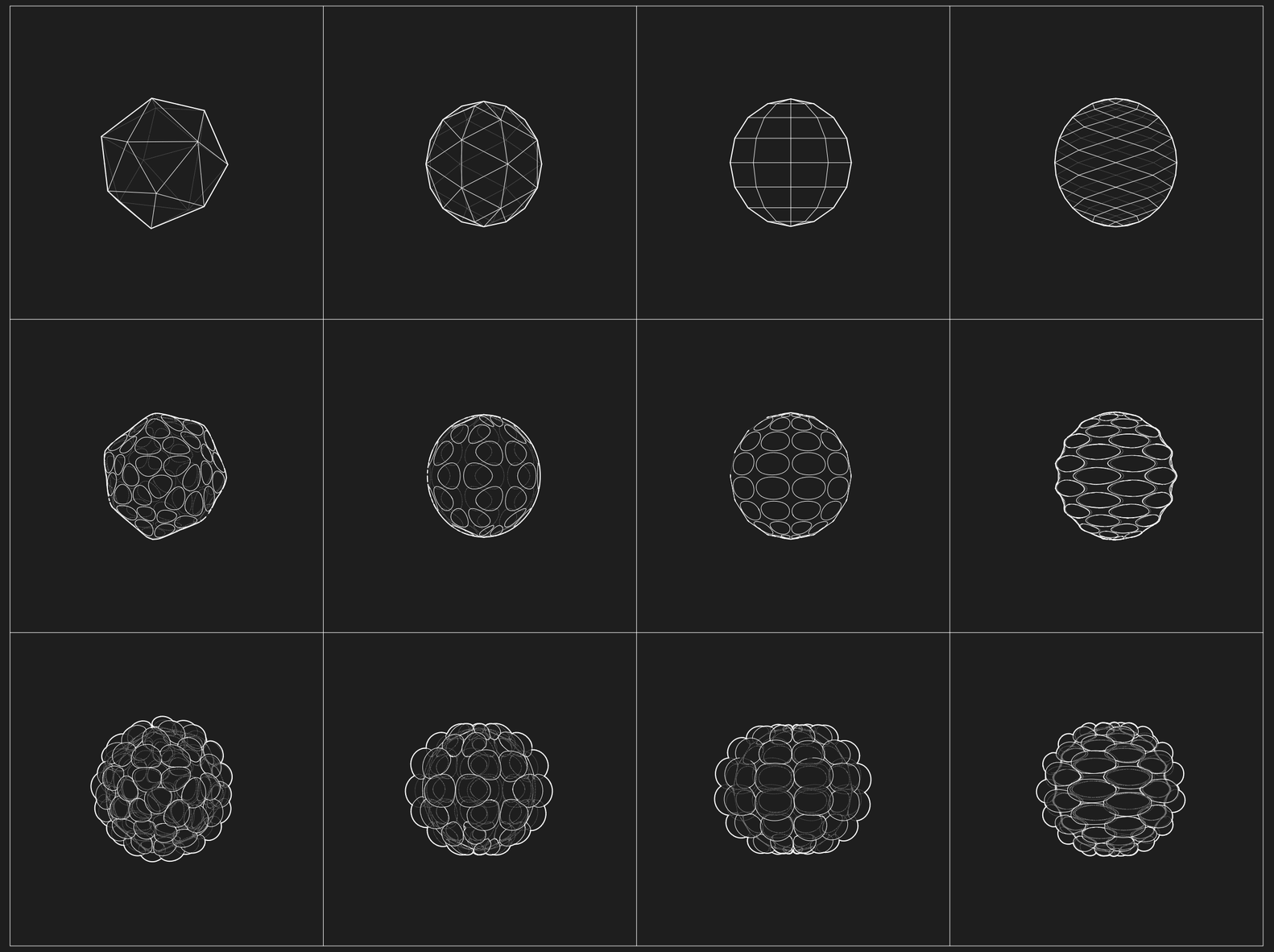
CREATURE'S SELF-EXPRESSION



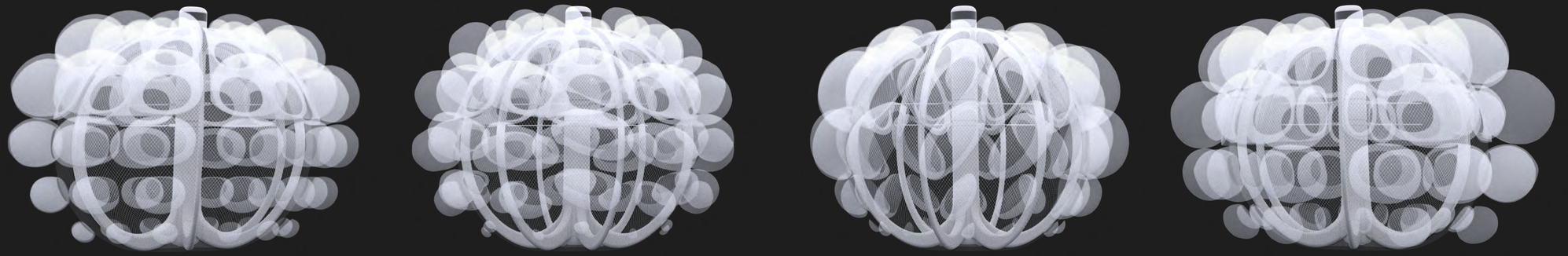
CONNECTION WITH OTHER CREATURES

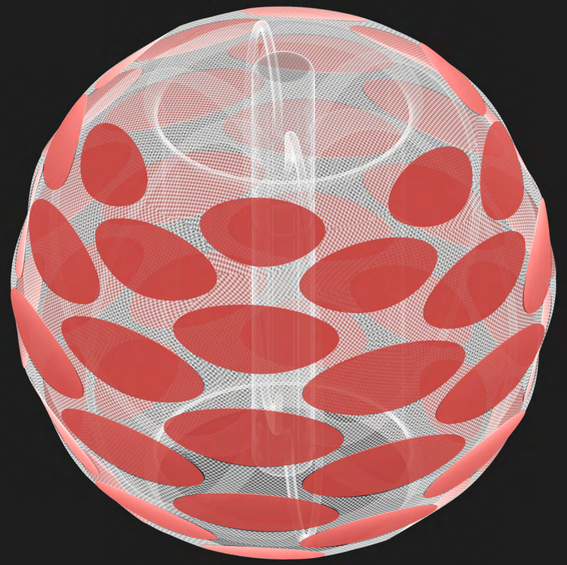


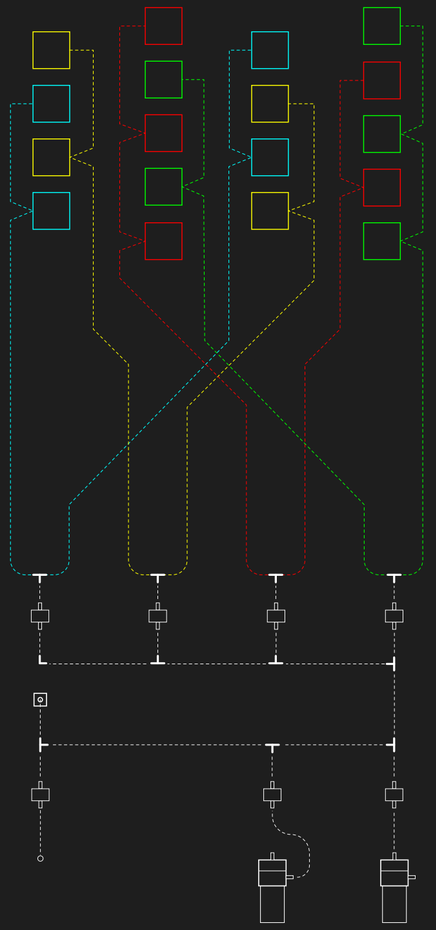
ENVIRONMENT MODULATION



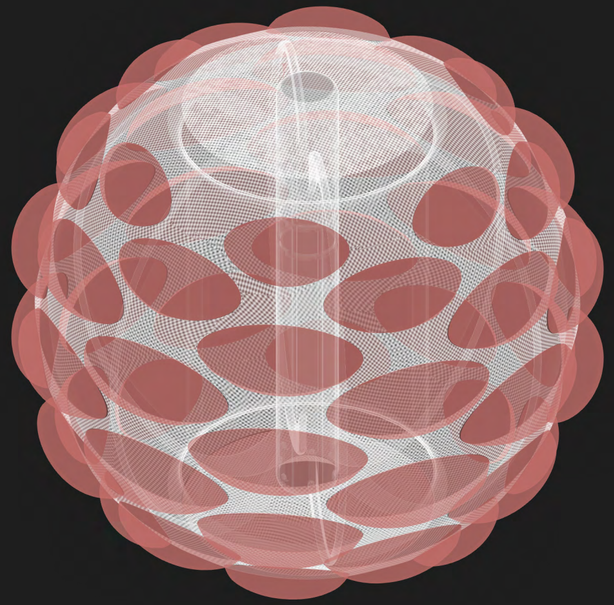
VARIATIONS: GRIDS

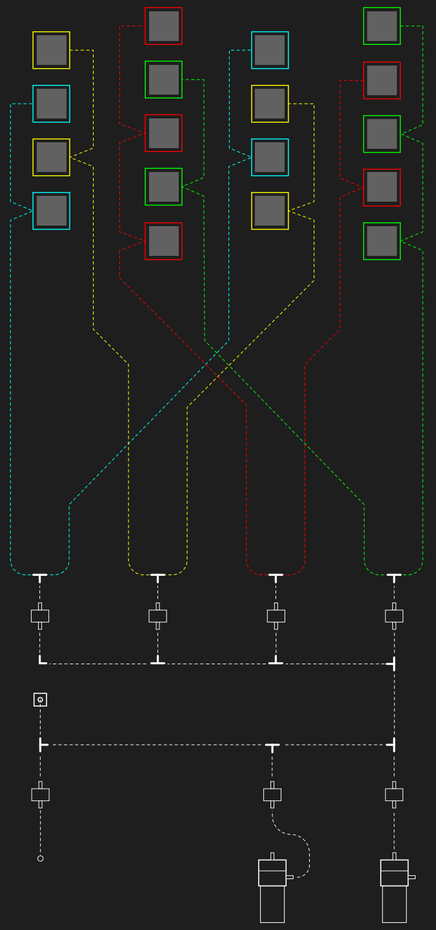




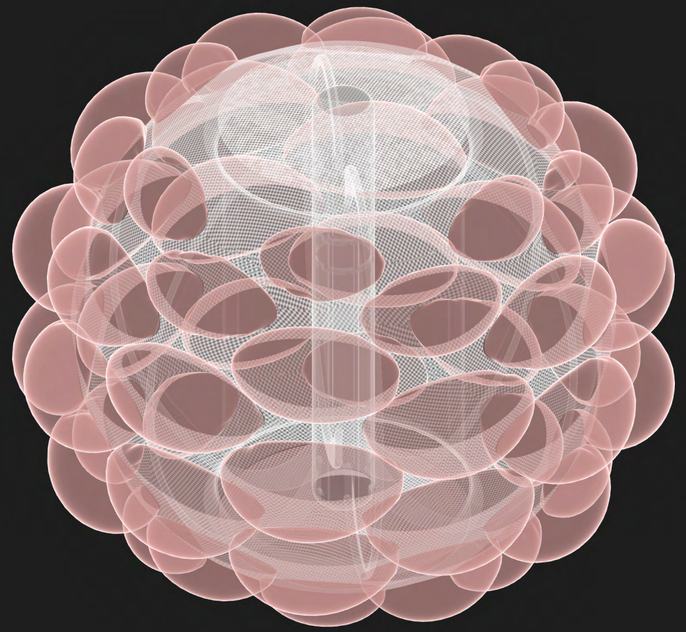


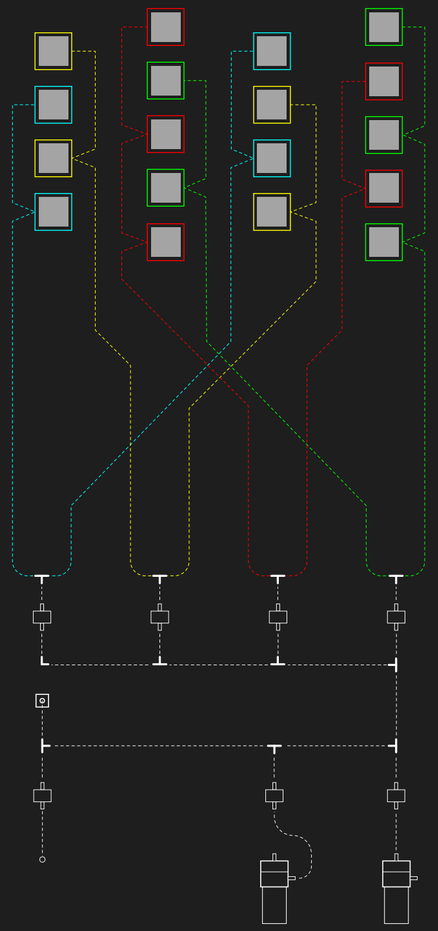
ACTUATION: INFLATION PATTERNS



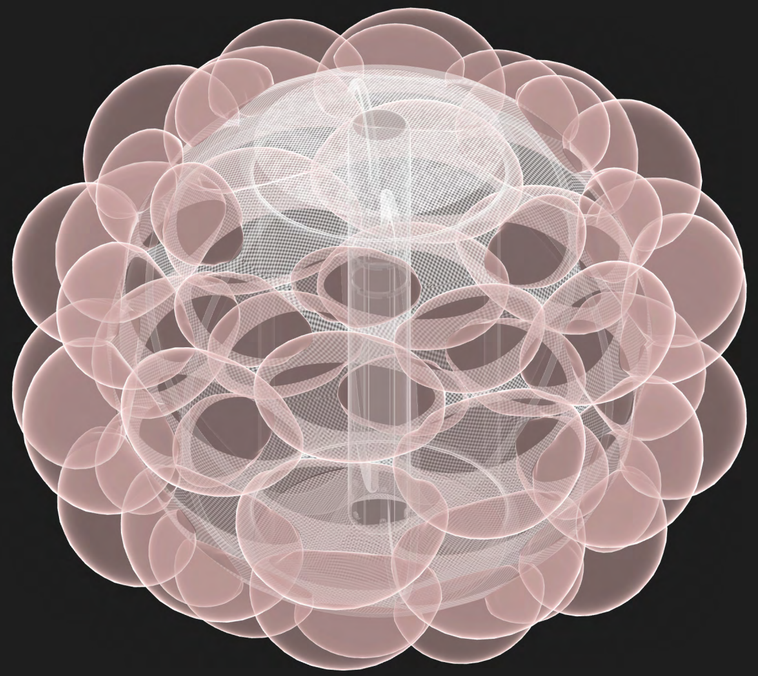


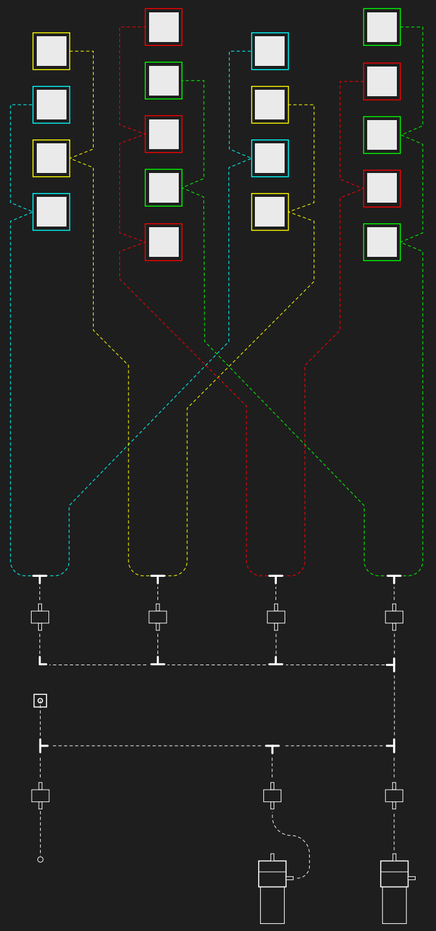
ACTUATION: INFLATION PATTERNS



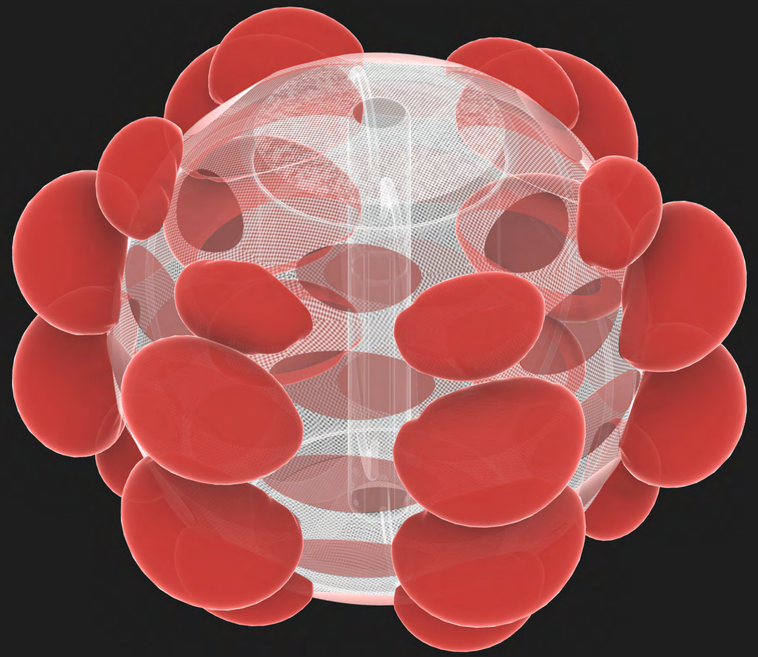


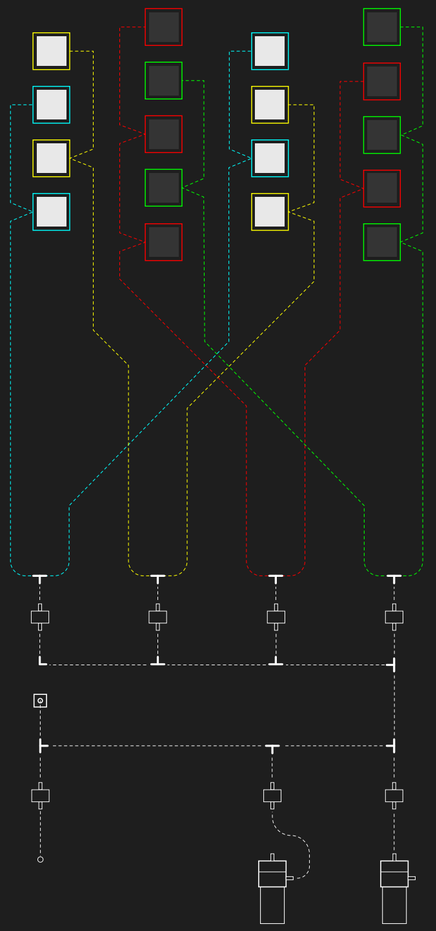
ACTUATION: INFLATION PATTERNS



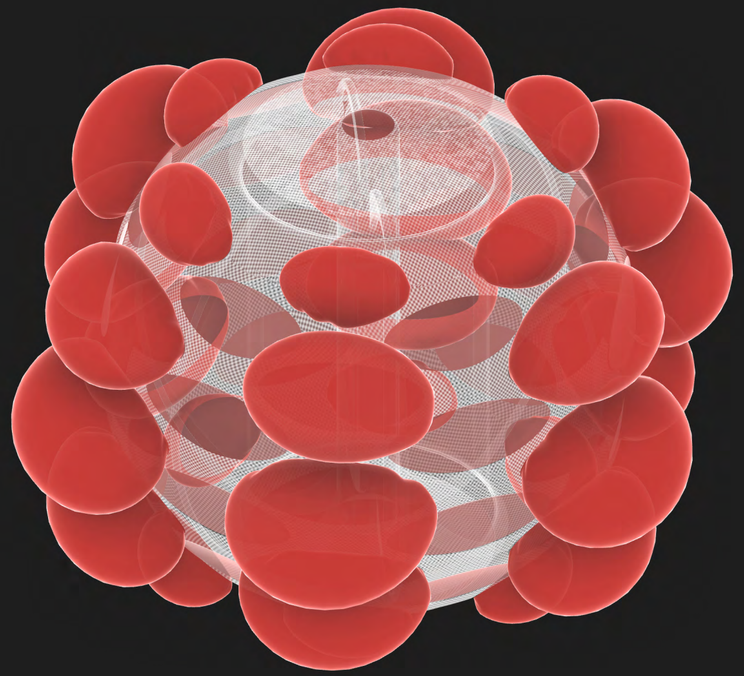


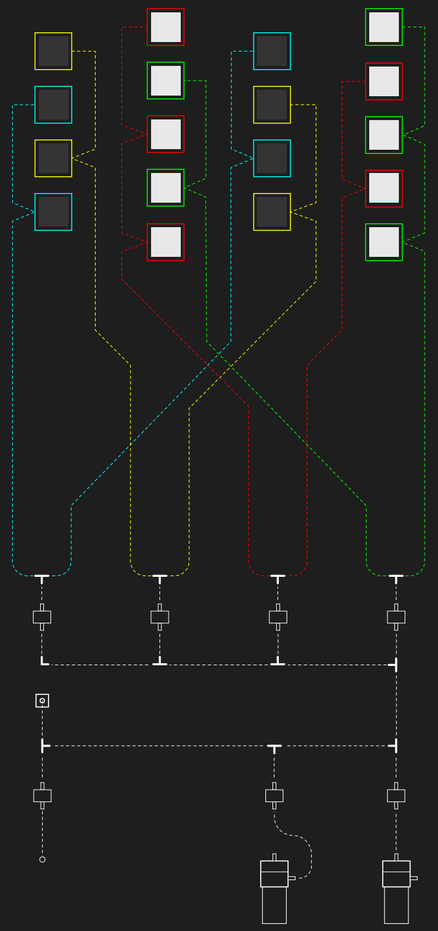
ACTUATION: INFLATION PATTERNS



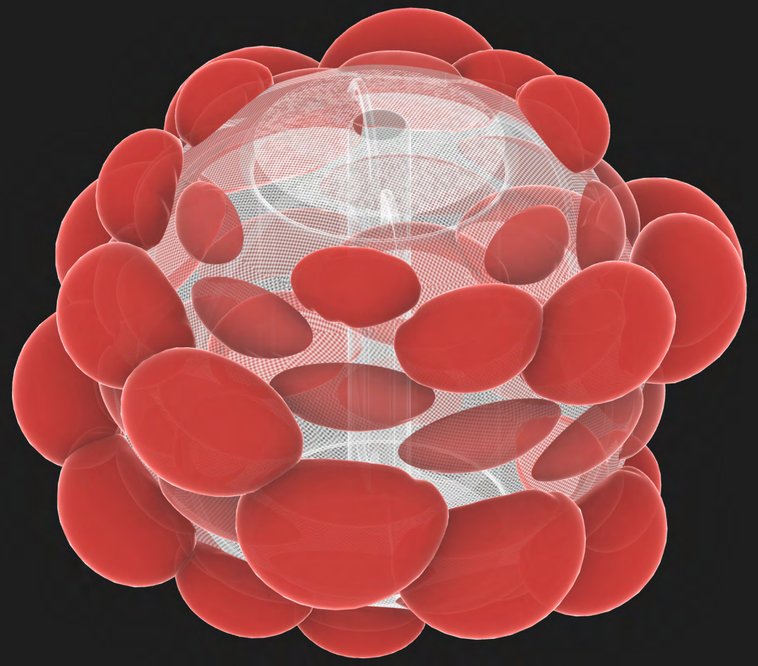


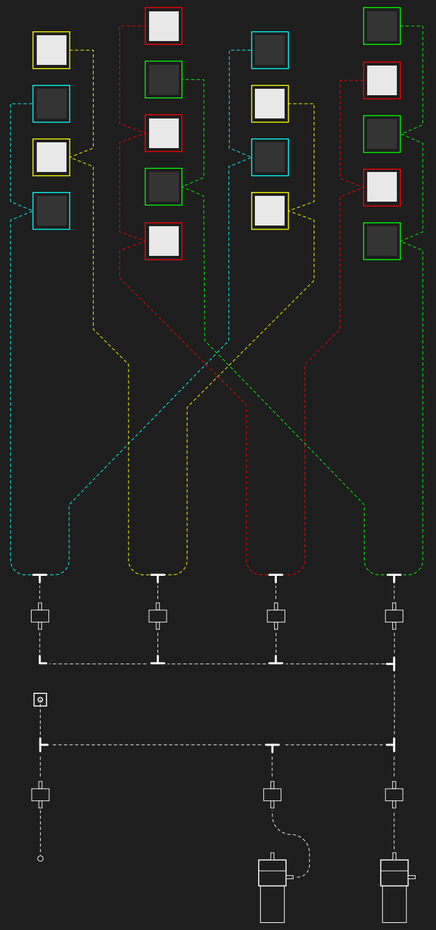
ACTUATION: INFLATION PATTERNS



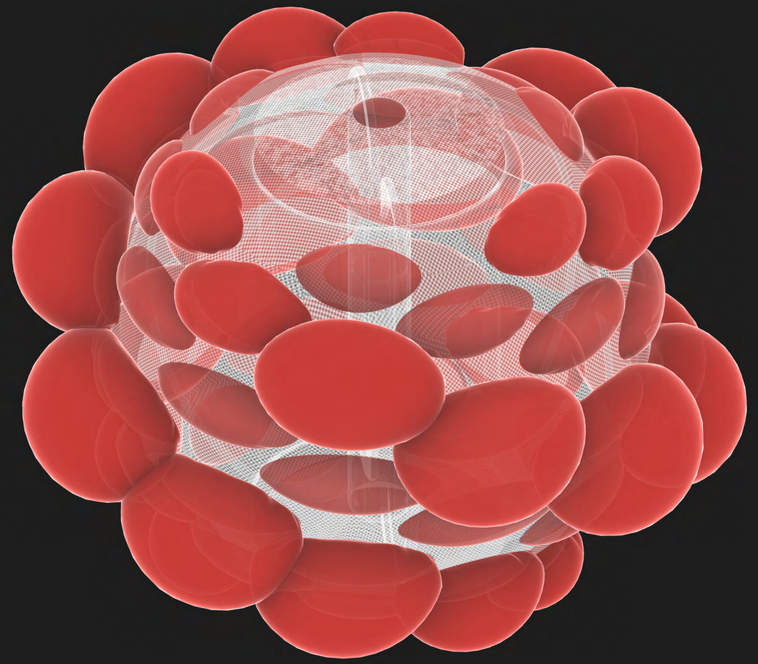


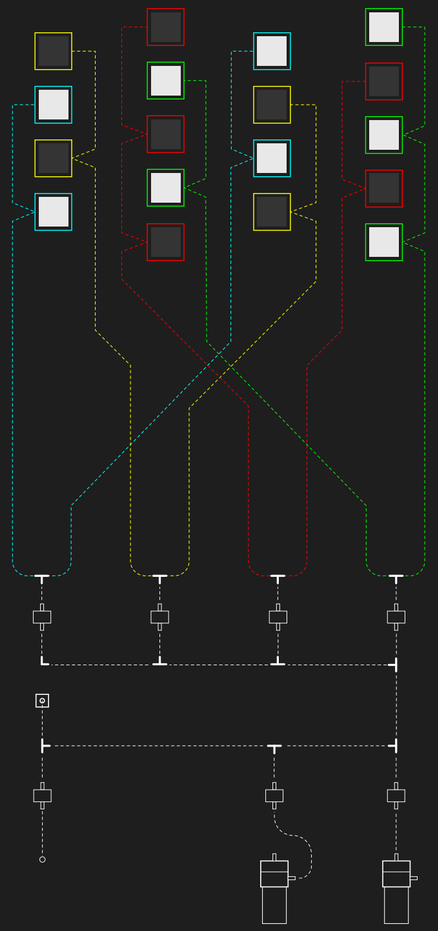
ACTUATION: INFLATION PATTERNS





ACTUATION: INFLATION PATTERNS





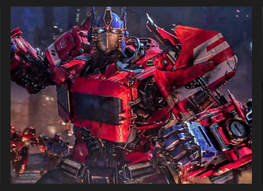
ACTUATION: INFLATION PATTERNS



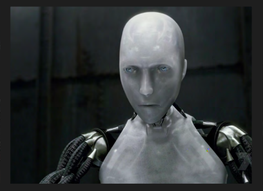
HAL 900
ISTJ



GLADOS
INSJ



OPTIMUS PRIME
ISFJ



SONNY
INFJ



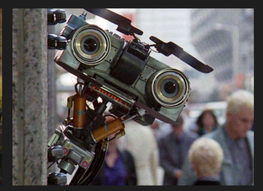
BENDER
ESTP



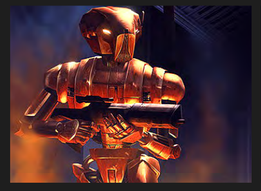
R2-D2
ENTP



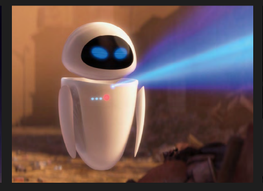
B.E.N.
ESFP



JOHNNY FIVE
ENFP



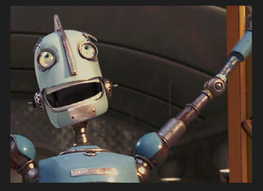
HK-47
ESTJ



EVE
ENSJ



C-3PO
ESFJ



HERTH COPPORBOTTOM
ENFJ



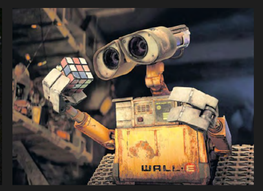
MARVIN
ISTP



BORG QUEEN
INTP



DATA
ISFP



WALL-E
INFP

MYERS BRIGGS PERSONALITY TYPES
FOR COMMON ROBOTS

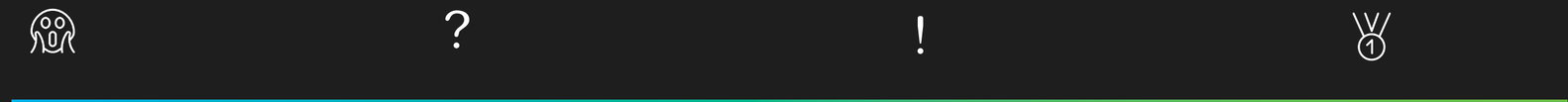
CREATURE'S SELF EXPRESSION

EMOTION SPECTRUM



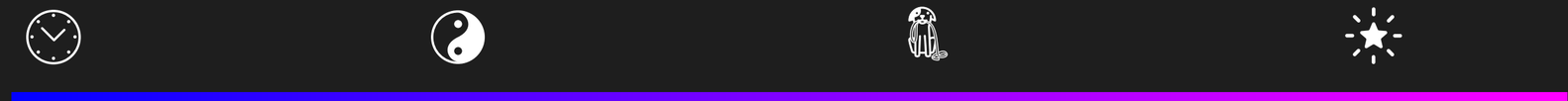
ANGRY | SAD | NEUTRAL | HAPPY

ATTENTION SPECTRUM



FEARFUL | CURIOUS | INTERESTED | TRUSTING

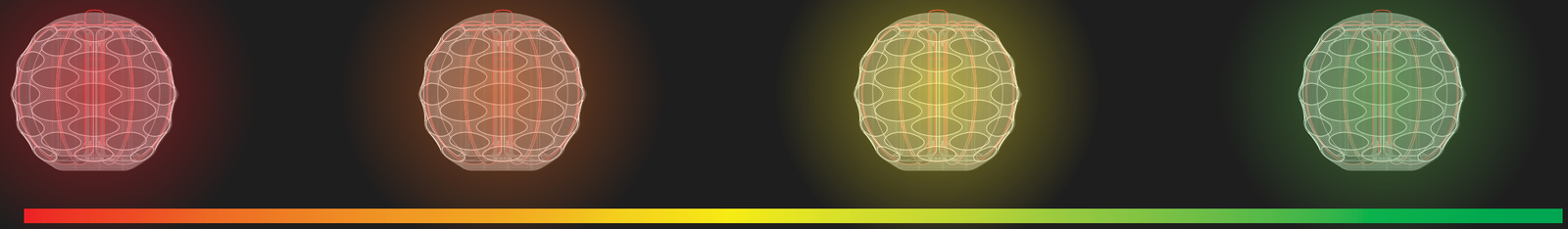
EXCITEMENT SPECTRUM



BORED | CALM | PLAYFUL | EXCITED

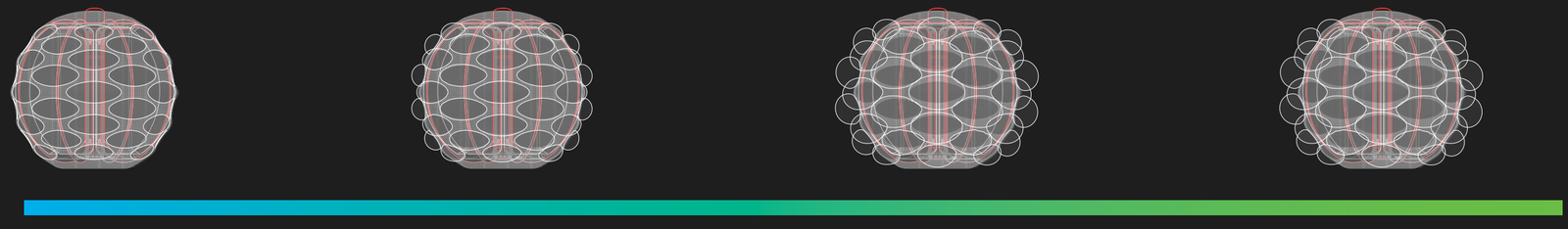
INTERACTIONS: EMOTIONAL ATTRIBUTES

EMOTION SPECTRUM



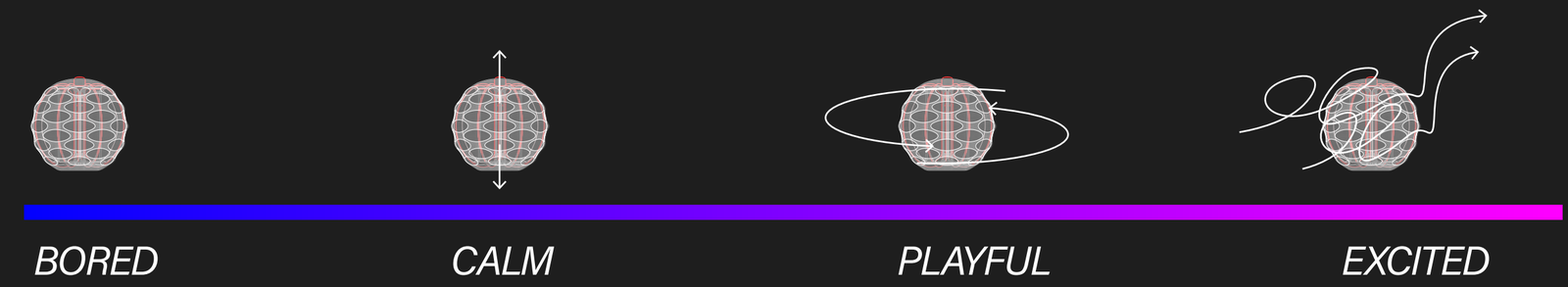
ANGRY SAD NEUTRAL HAPPY

ATTENTION SPECTRUM



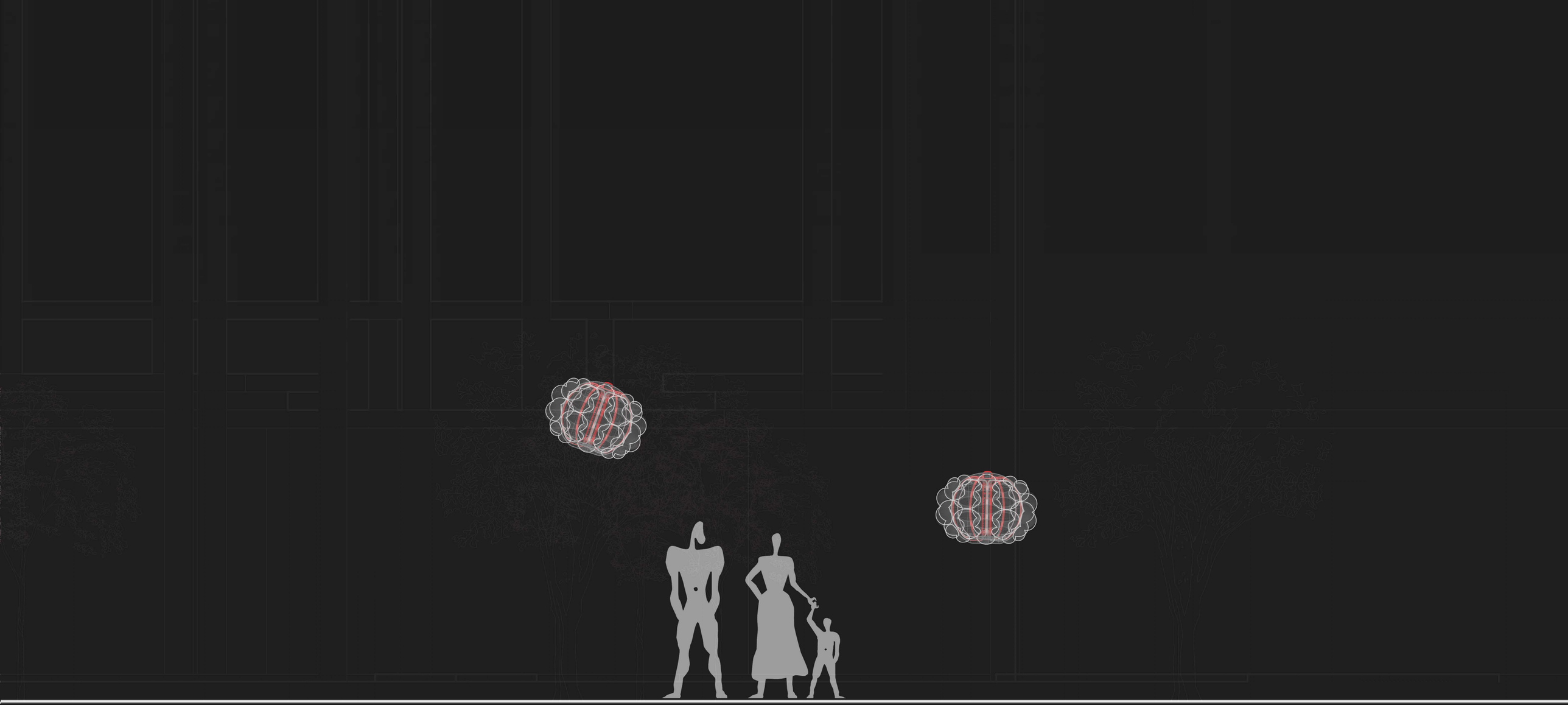
FEARFUL CURIOUS INTERESTED TRUSTING

EXCITEMENT SPECTRUM

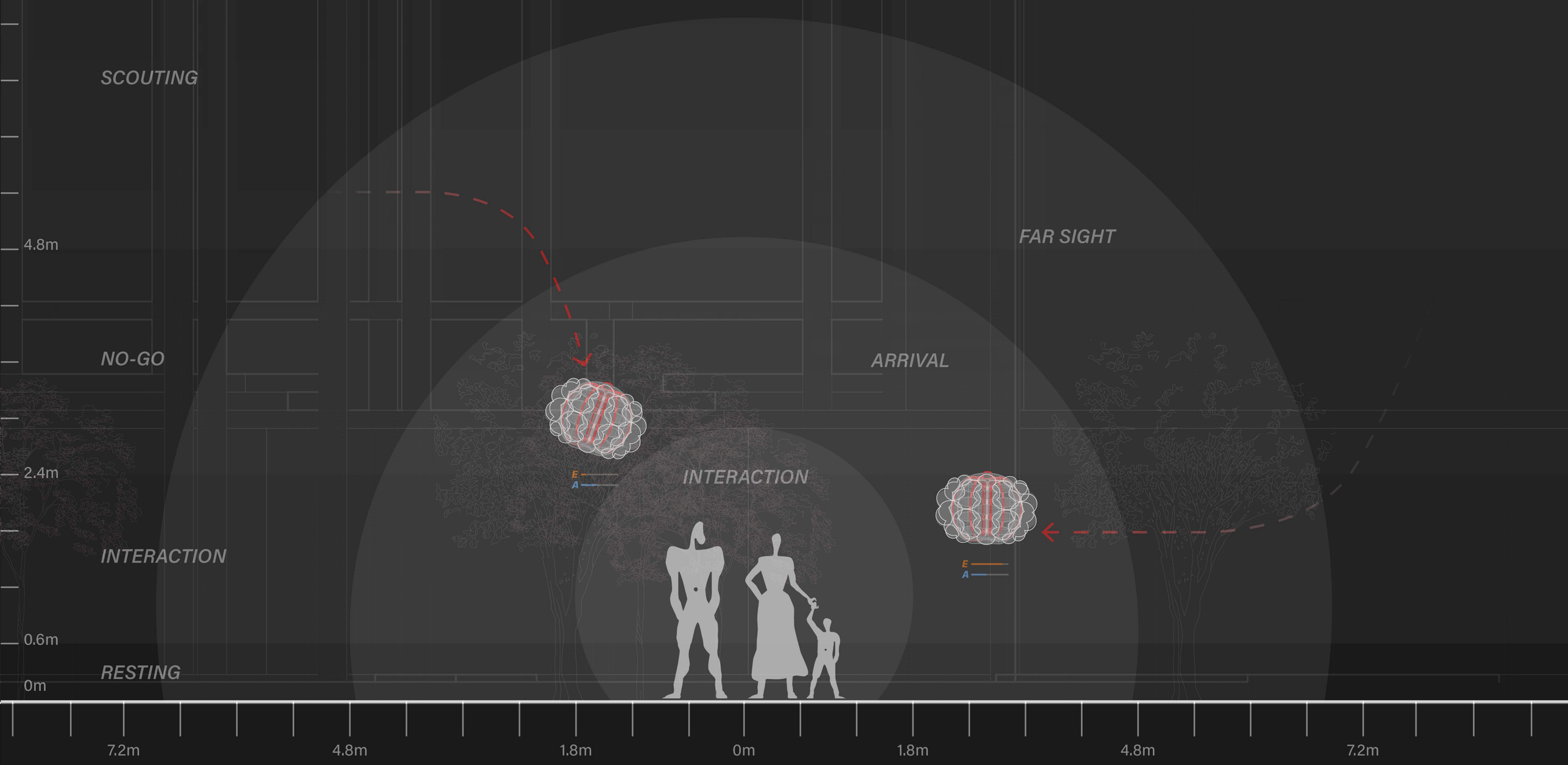


BORED CALM PLAYFUL EXCITED

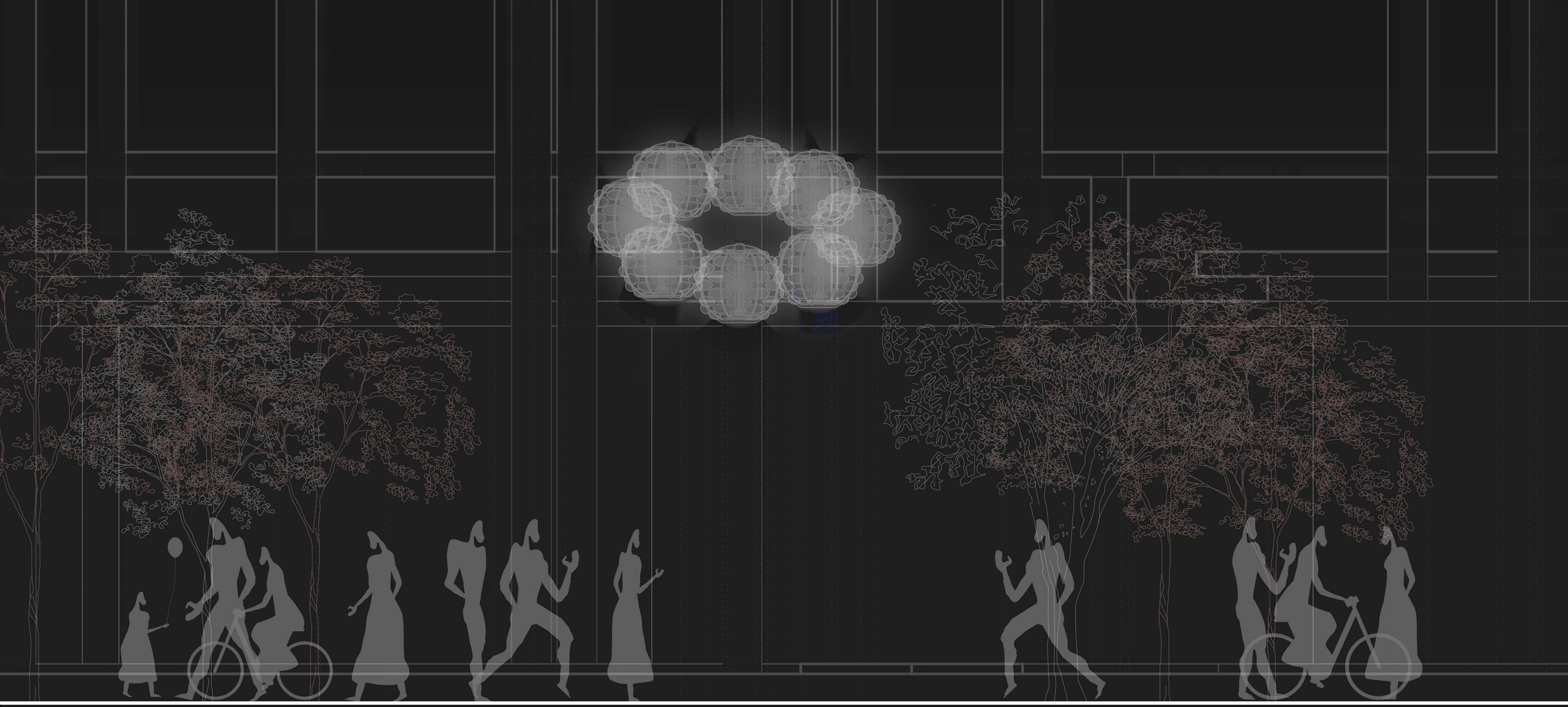
INTERACTIONS: EMOTIONAL ATTRIBUTES



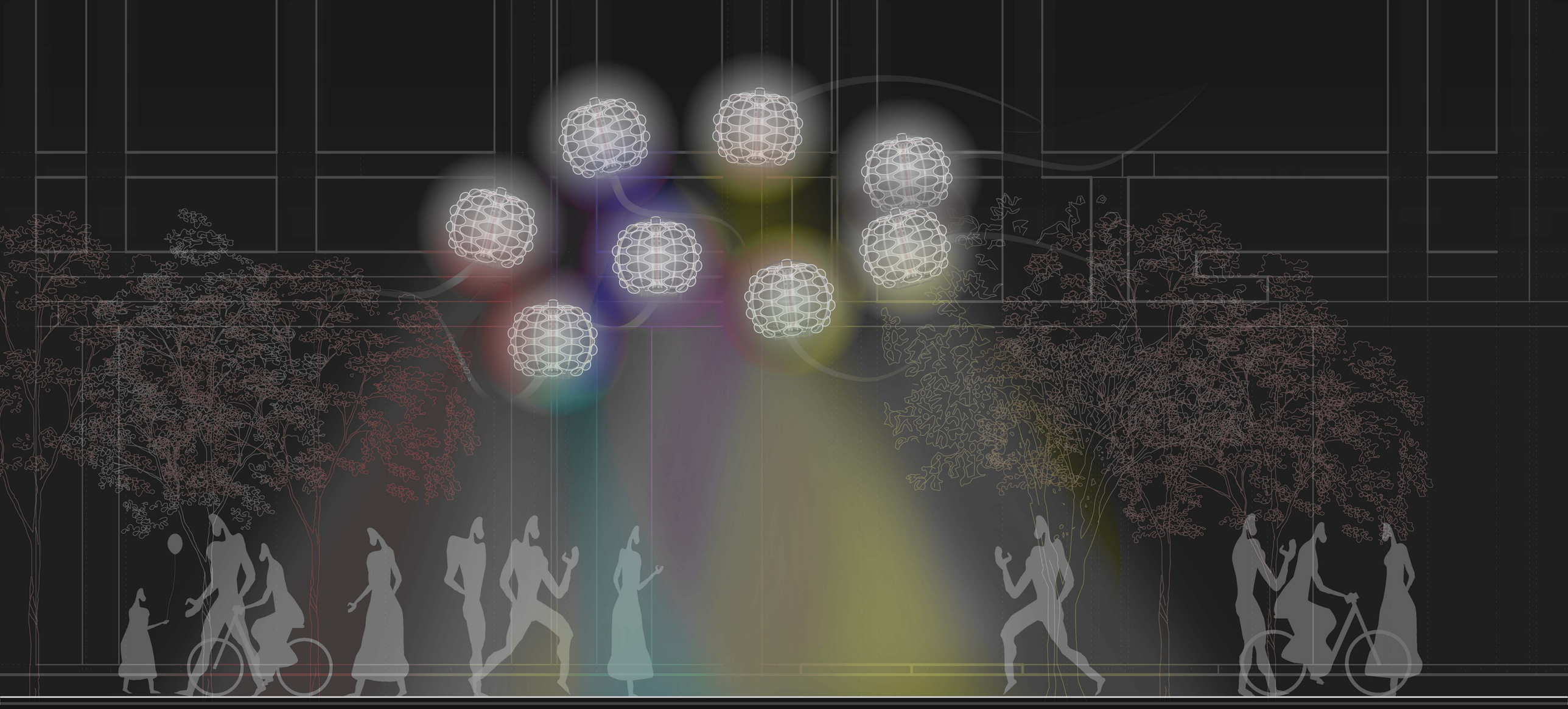
INTERACTIONS: EMOTIONAL ATTRIBUTES



INTERACTIONS: EMOTIONAL ATTRIBUTES

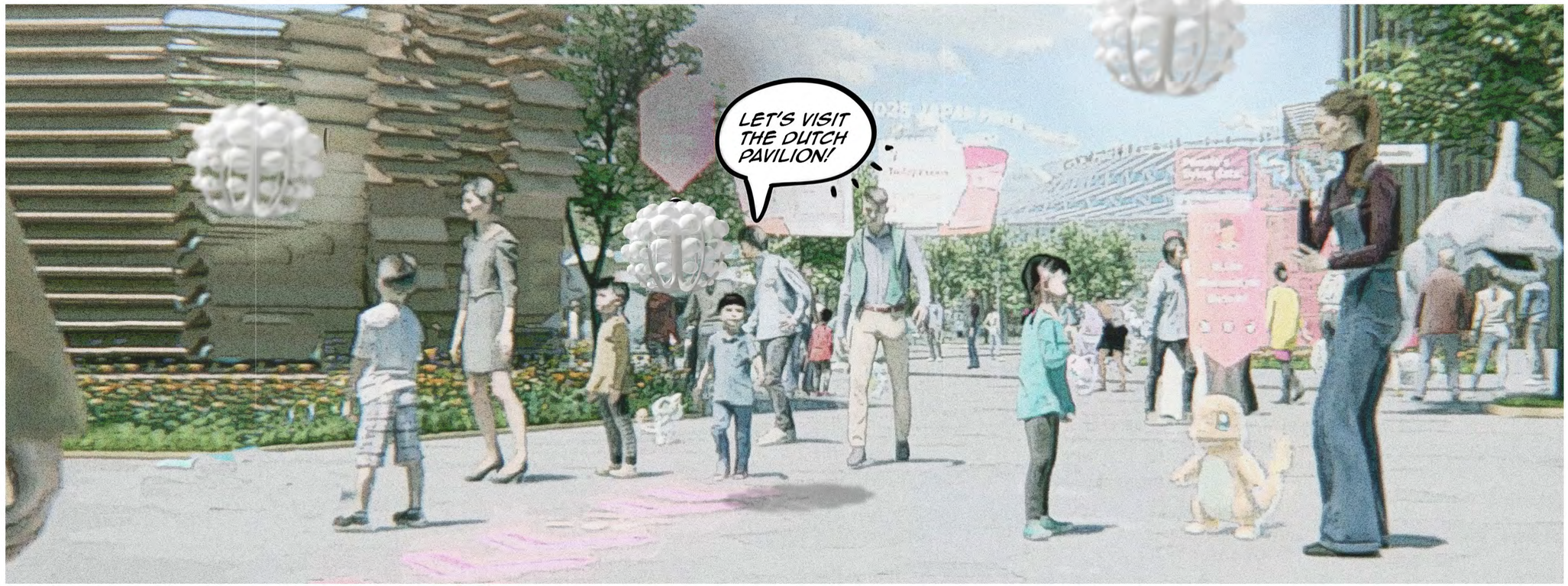


INTERACTIONS: MARKING A PLACE

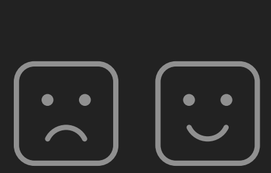


AGITATED

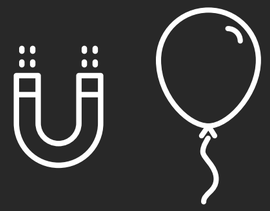
INTERACTIONS: MARKING A PLACE


LET'S VISIT THE DUTCH PAVILION!

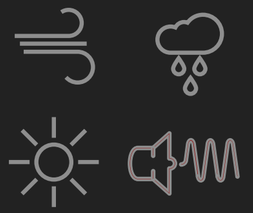
INFLATABLE SKIN



CREATURE'S SELF-EXPRESSION



CONNECTION WITH OTHER CREATURES

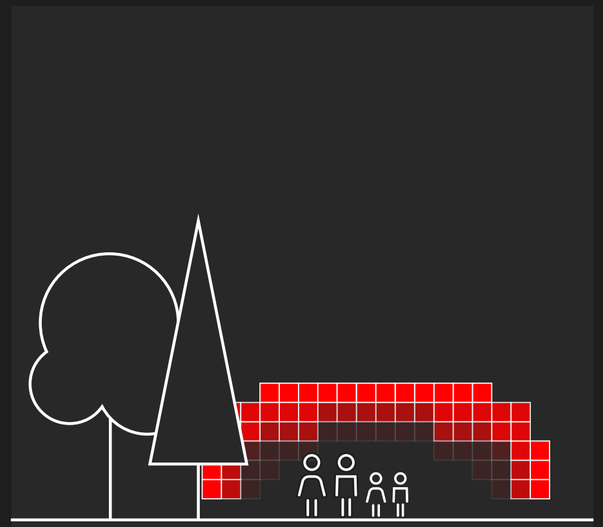


ENVIRONMENT MODULATION

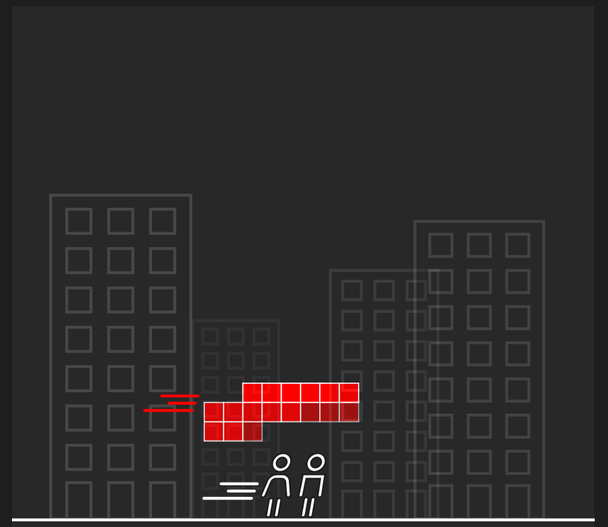




ADAPTS TO
URBAN LANDSCAPE

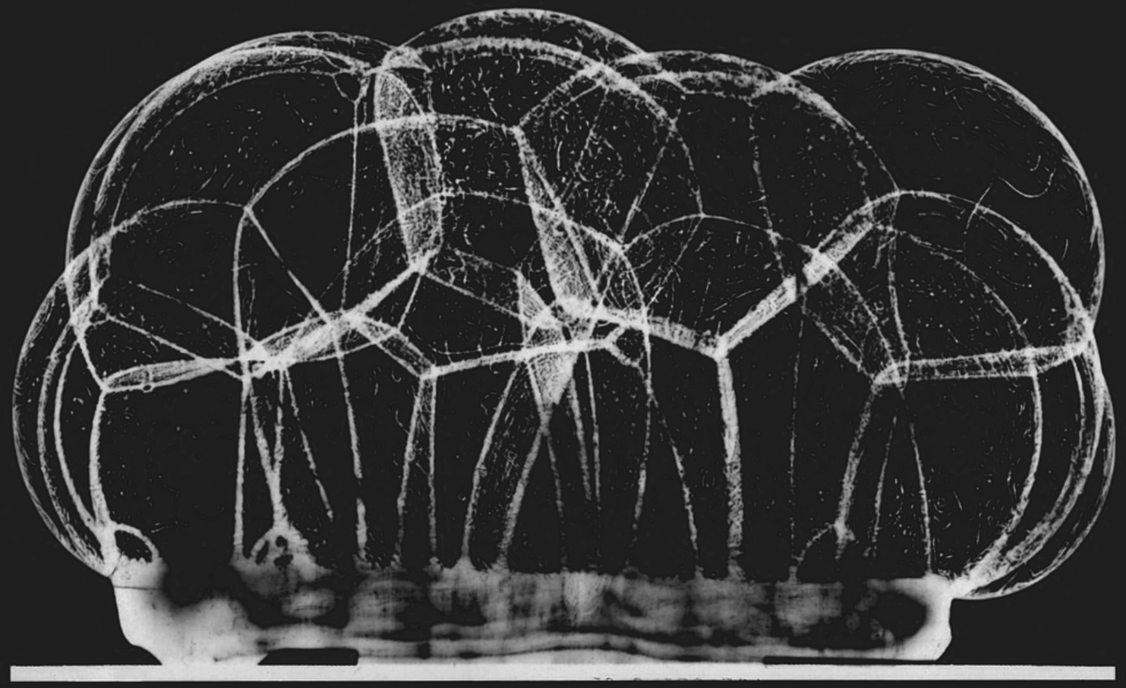


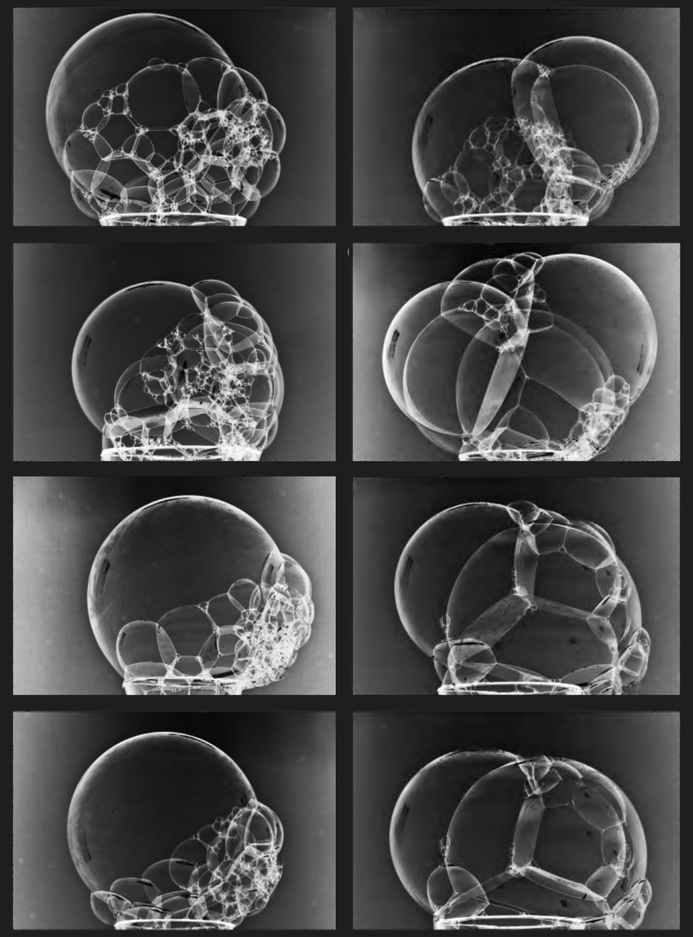
STATIONARY
IN PUBLIC SPACES



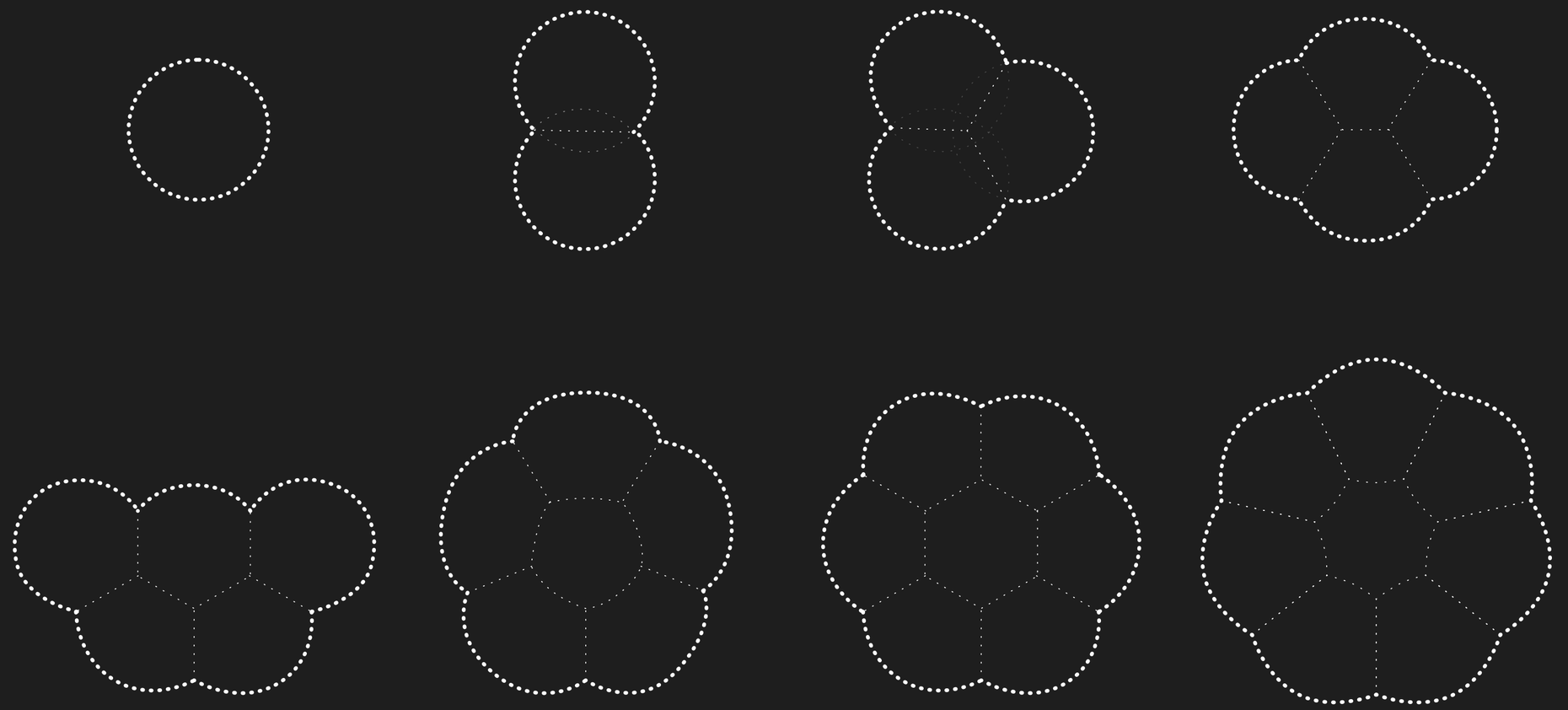
MOVES
THROUGH THE CITY

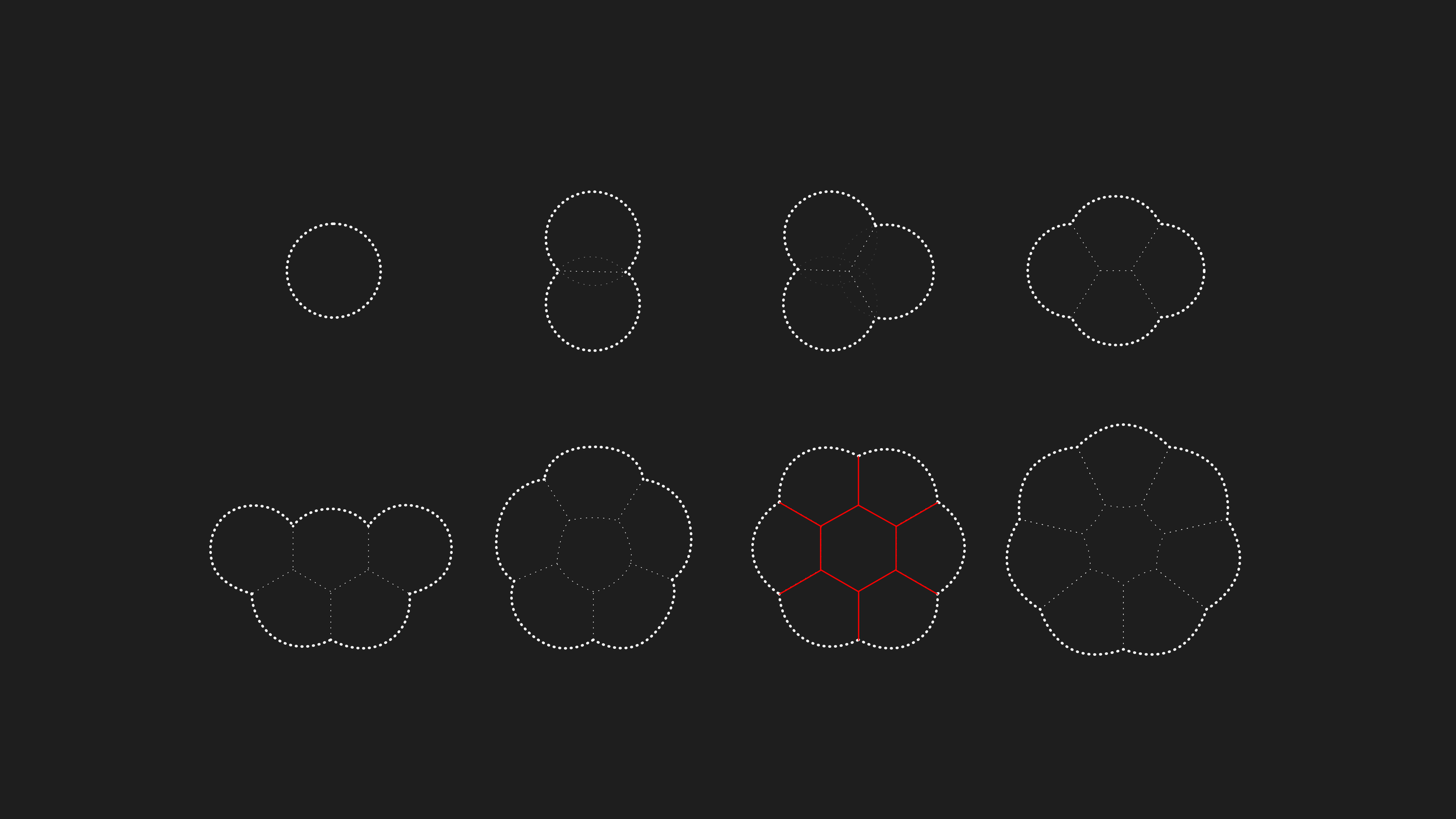
ASSEMBLY: TYPES AND IMPLICATIONS

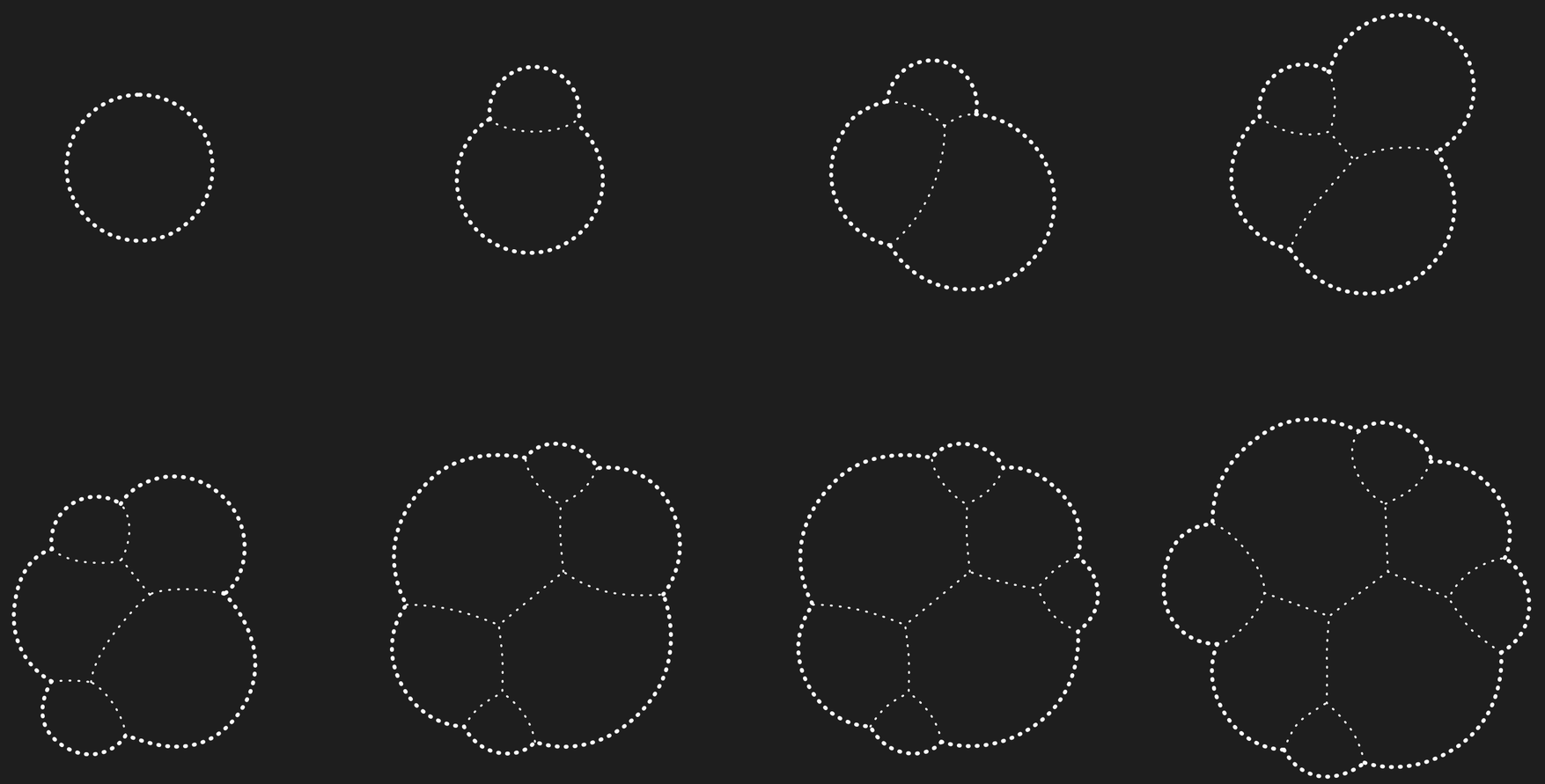




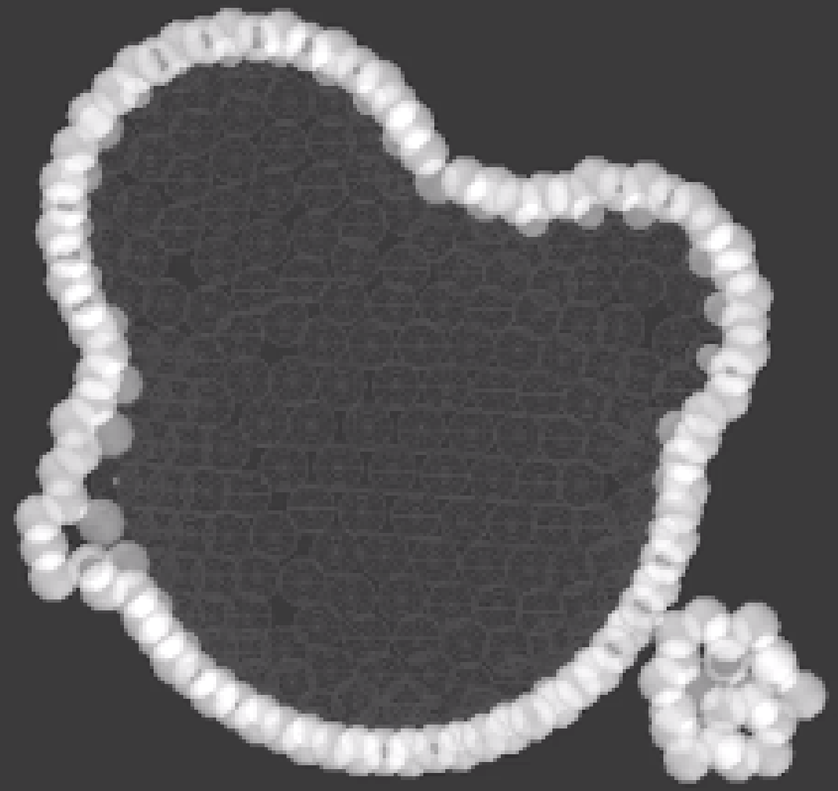
FREI OTTO, SOAP BUBBLE EXPERIMENTS, 1962

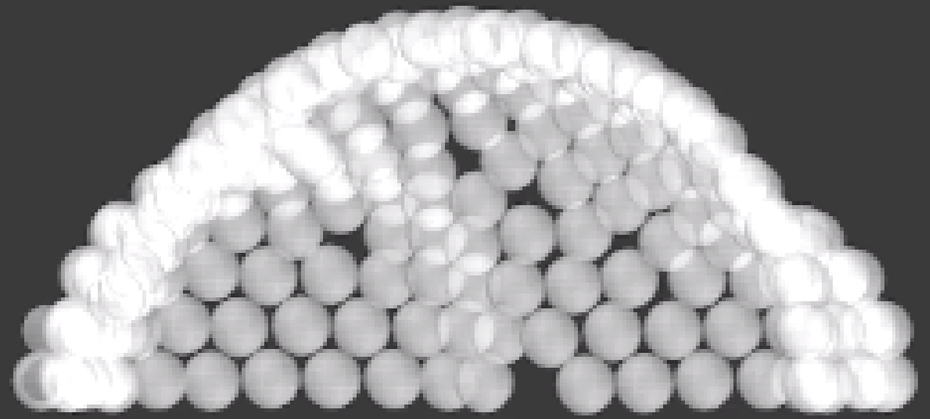


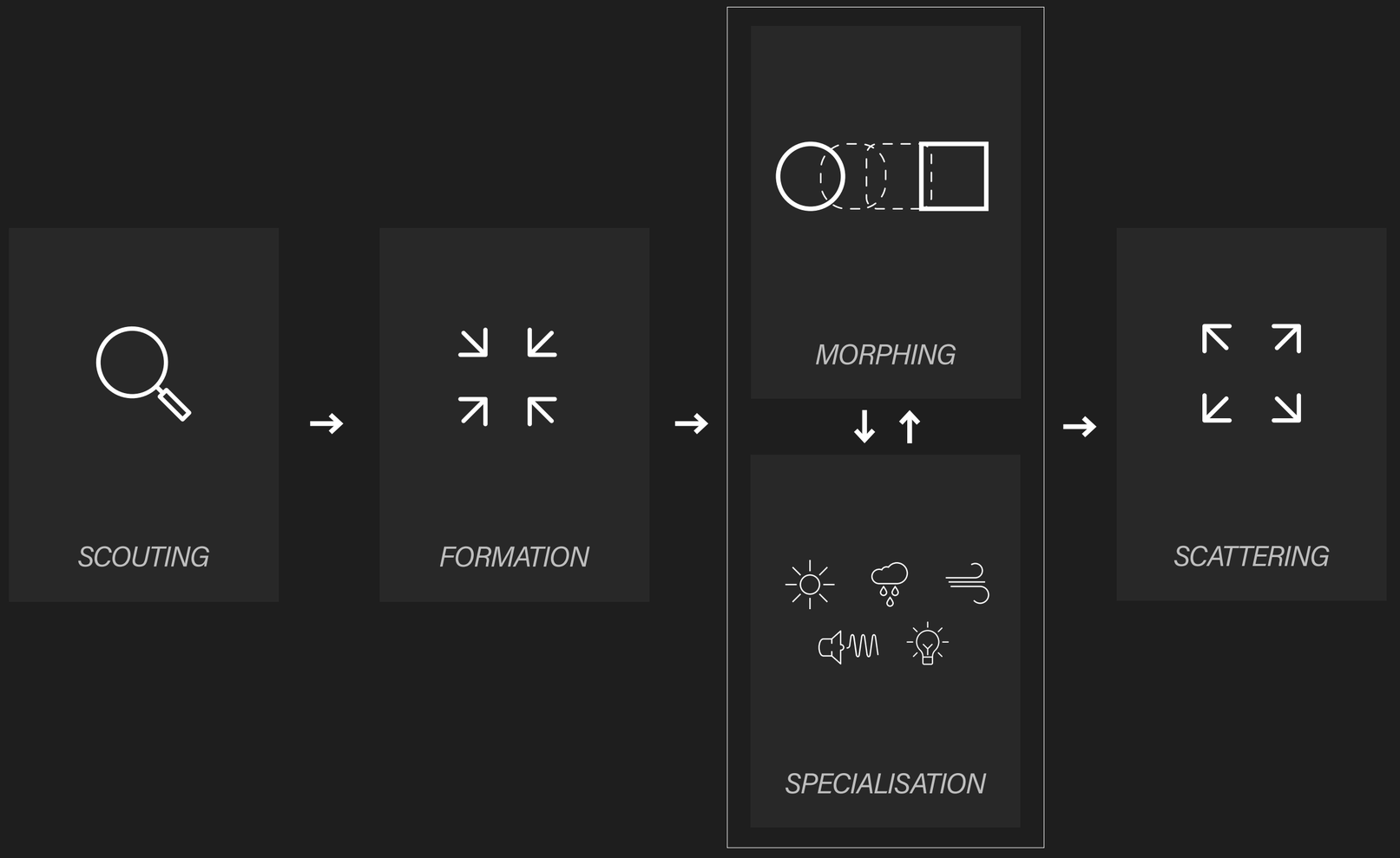




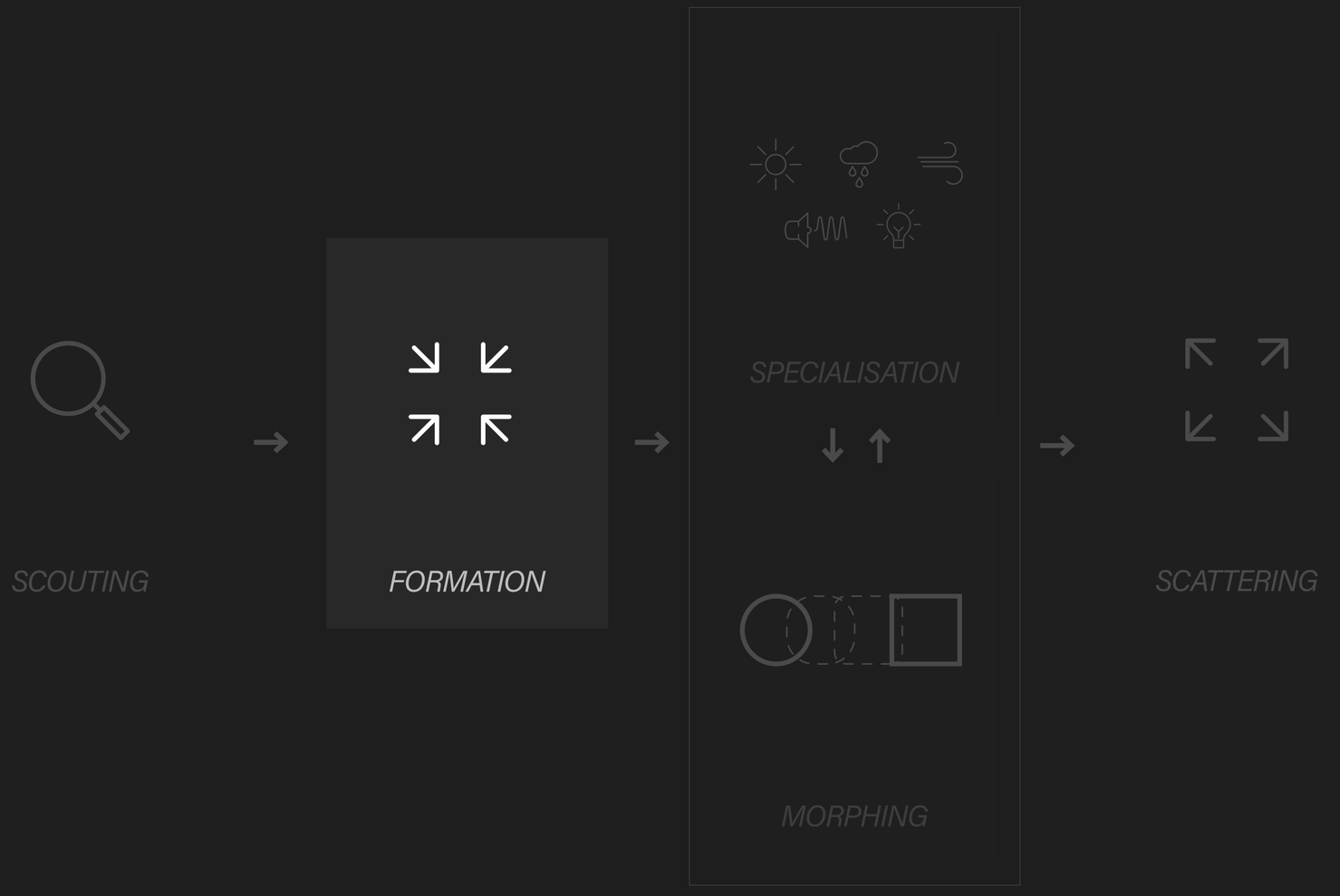




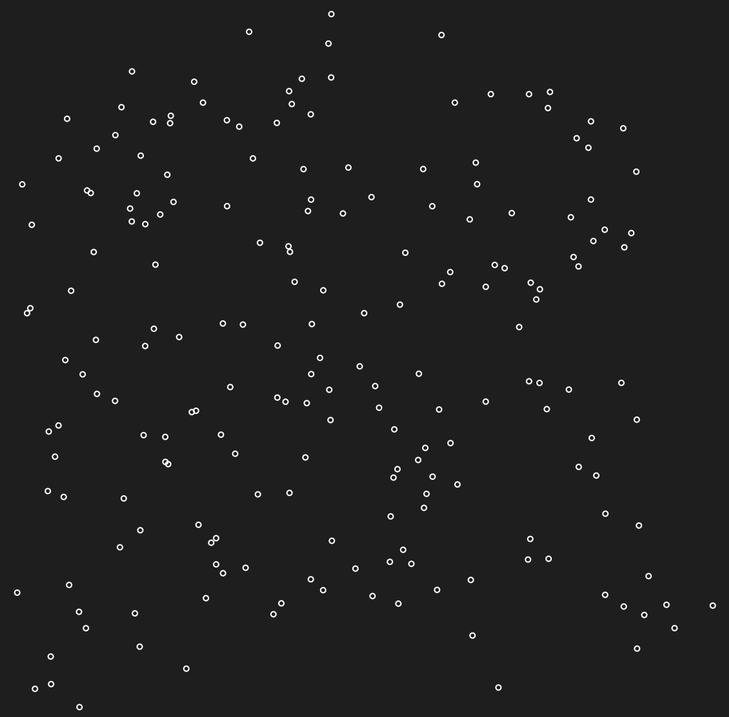


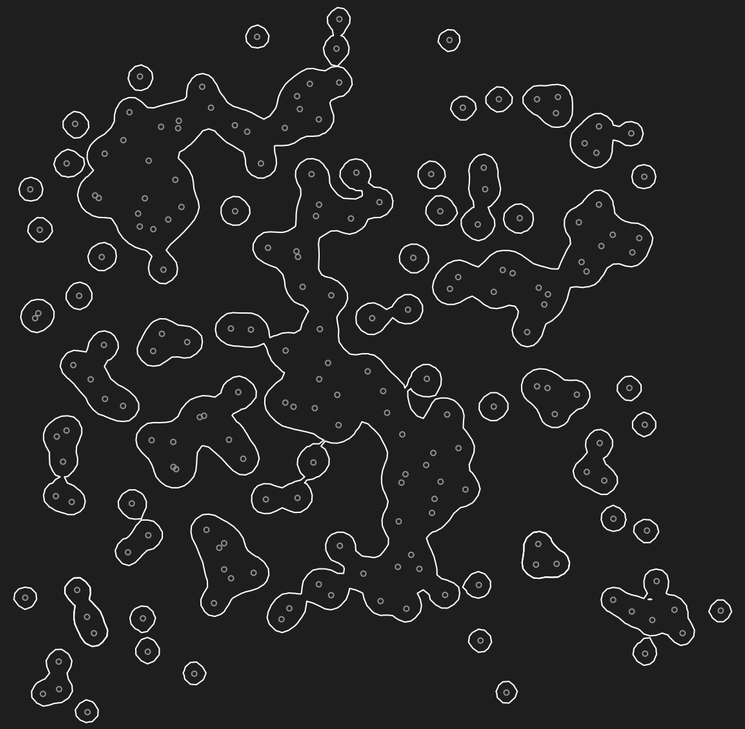


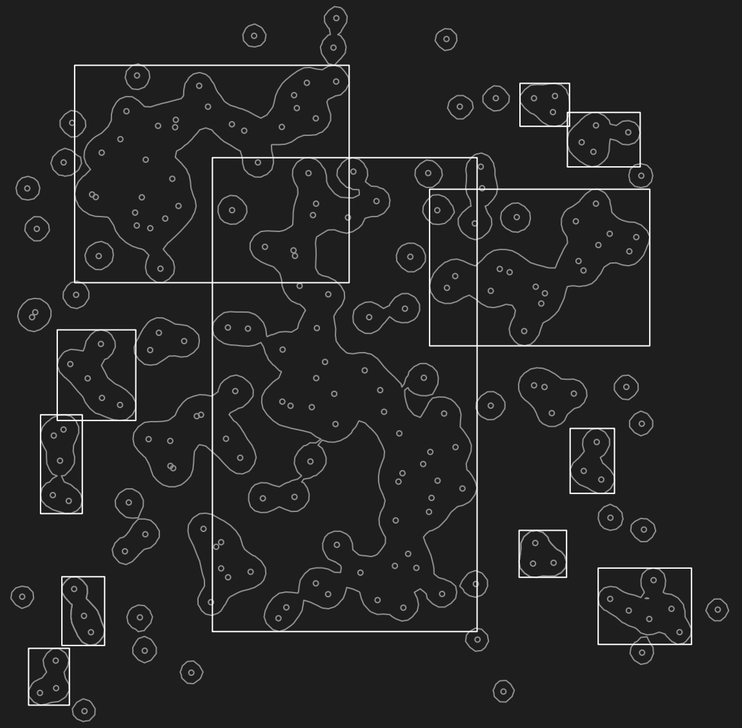
ASSEMBLY: FORMATION PROCESS

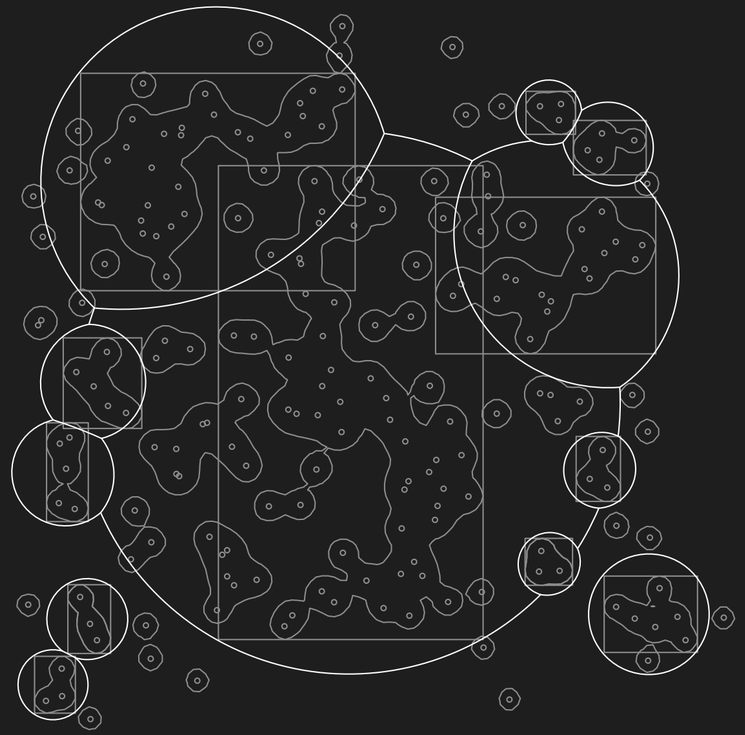


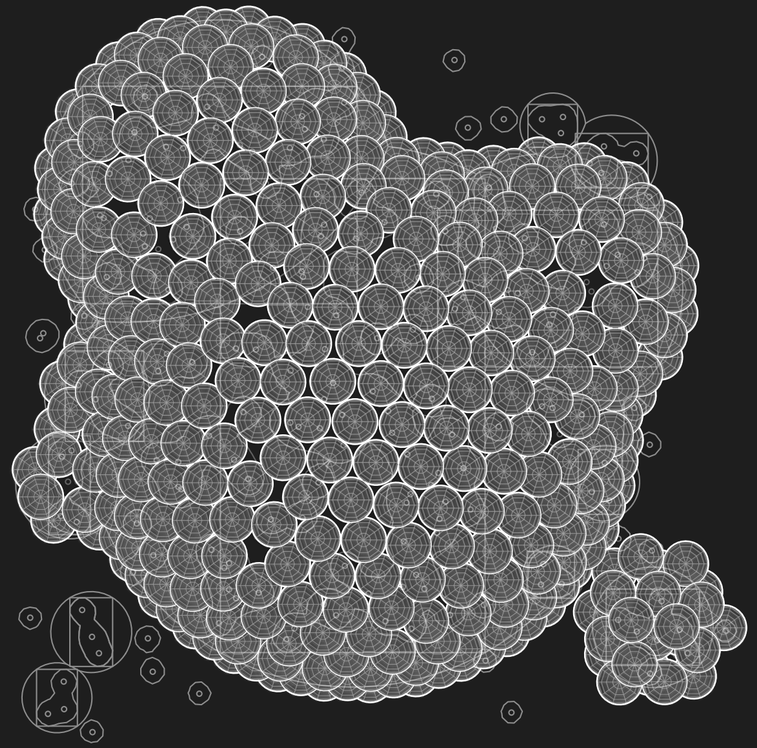
ASSEMBLY: FORMATION PROCESS









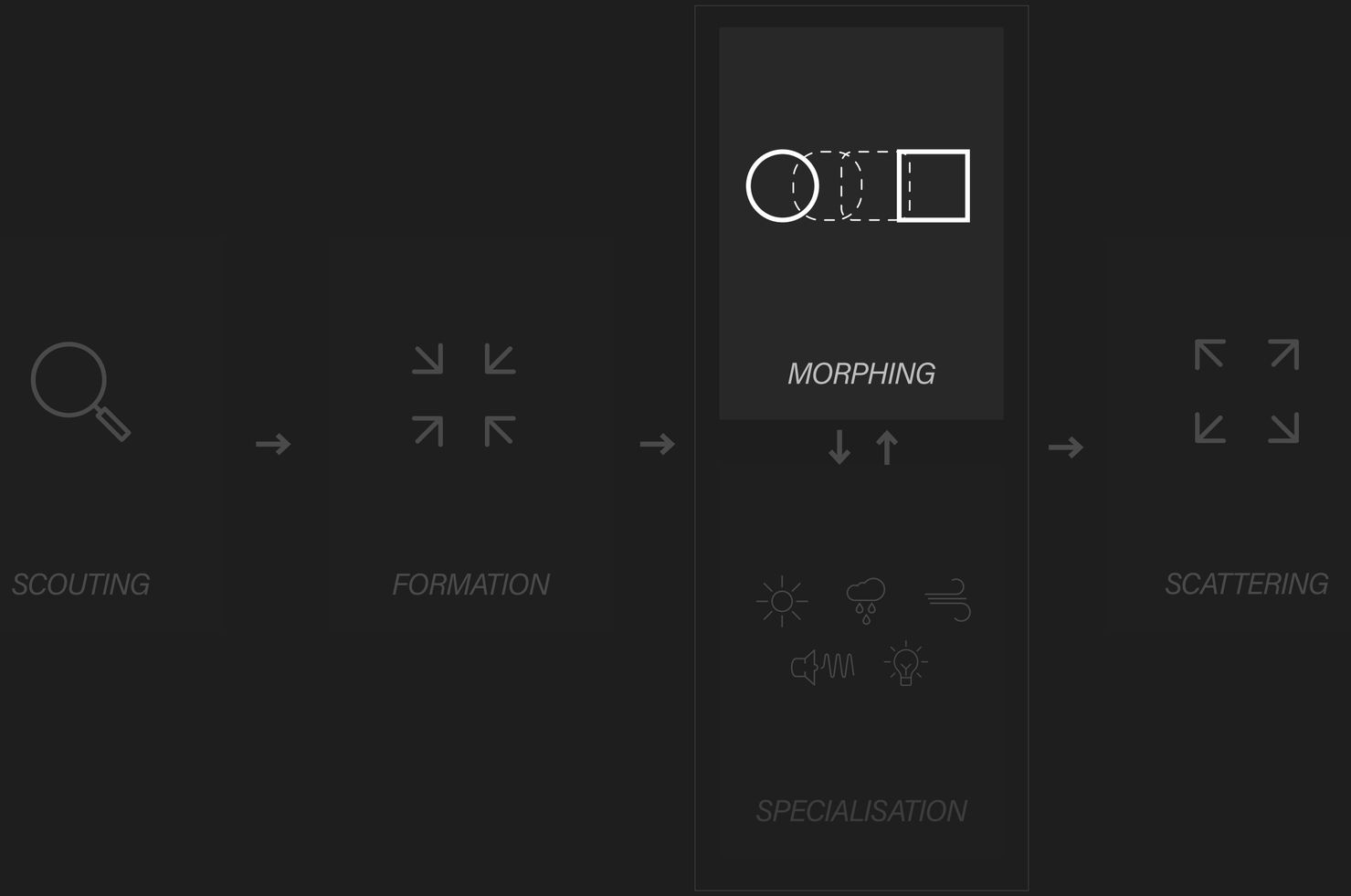




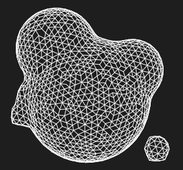


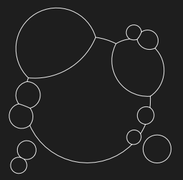


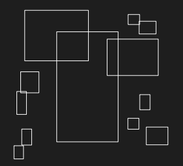


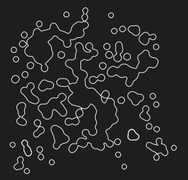


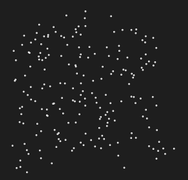
ASSEMBLY: MORPHING PROCESS

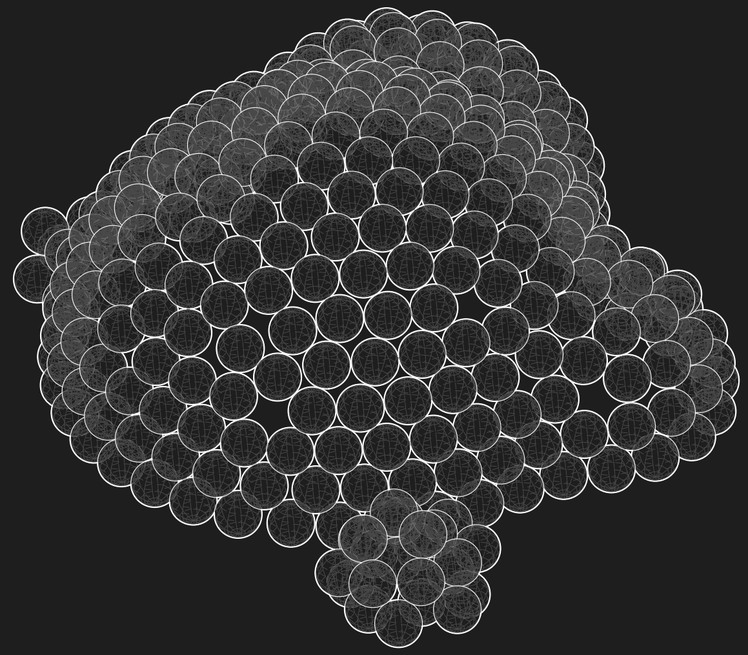




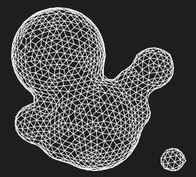


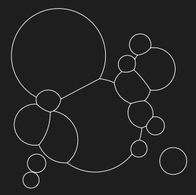


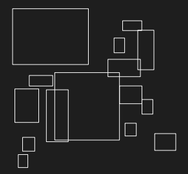


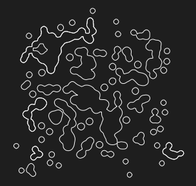


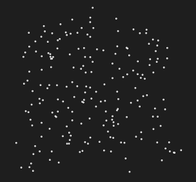
ASSEMBLY FROM CROWD SNAPSHOT #1

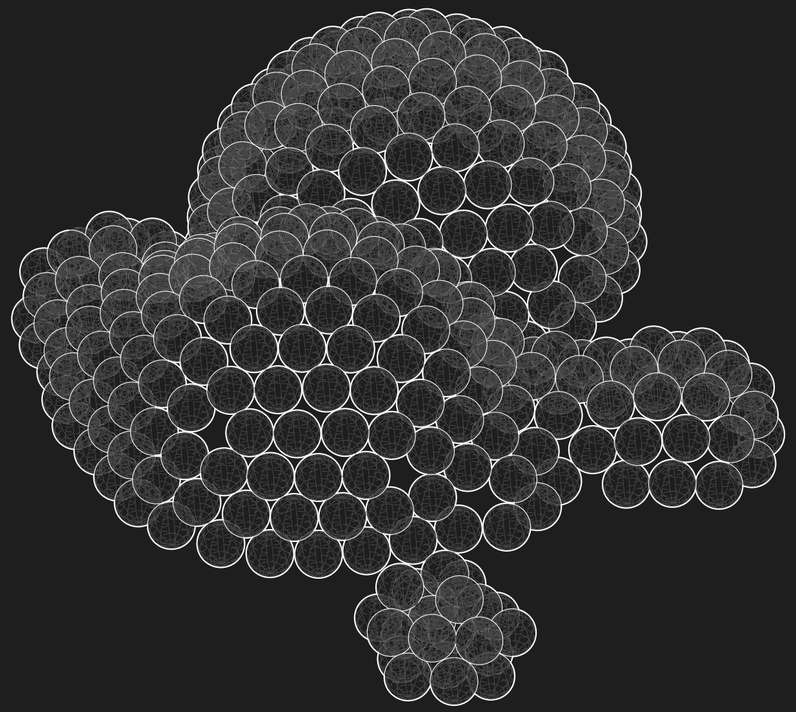








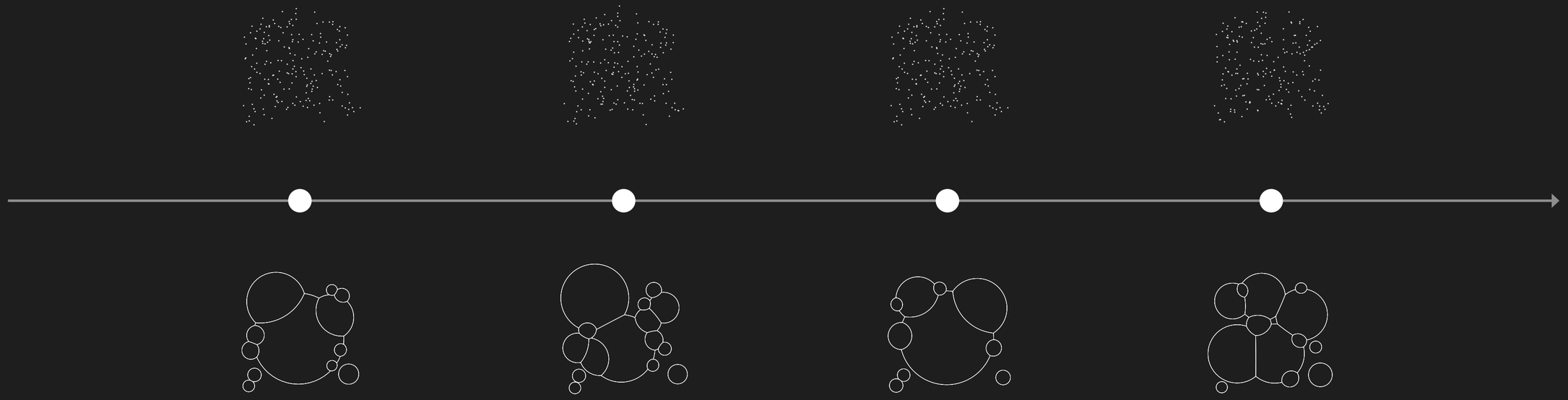




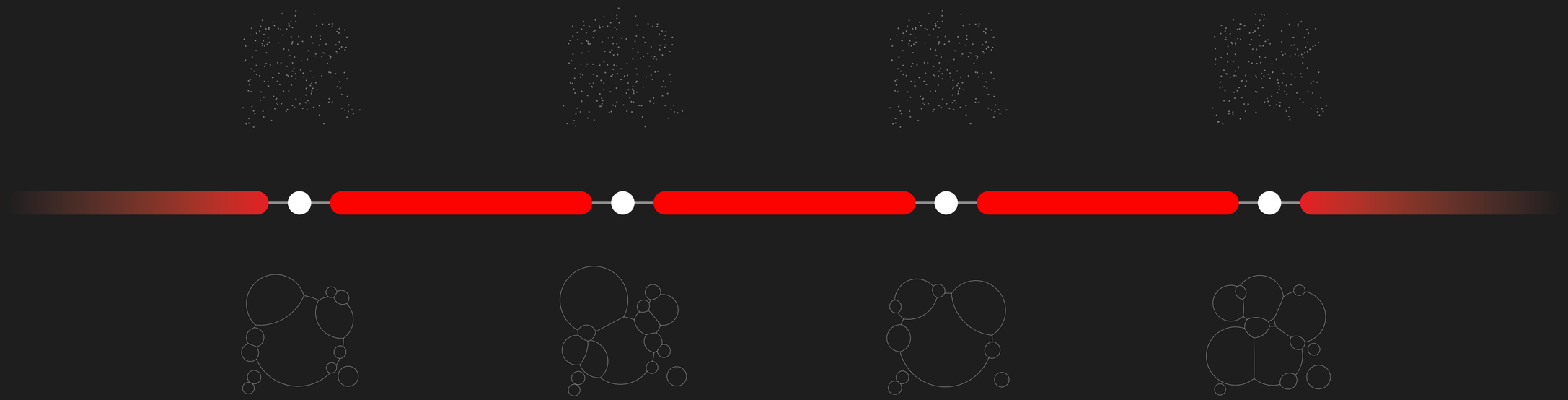
ASSEMBLY FROM CROWD SNAPSHOT #2



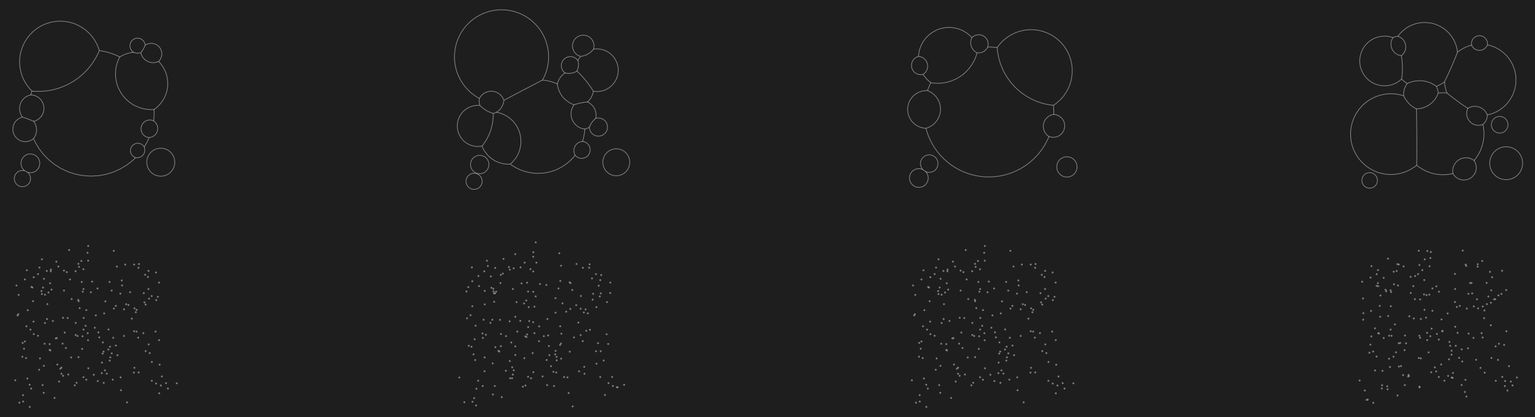
CROWD SNAPSHOTS

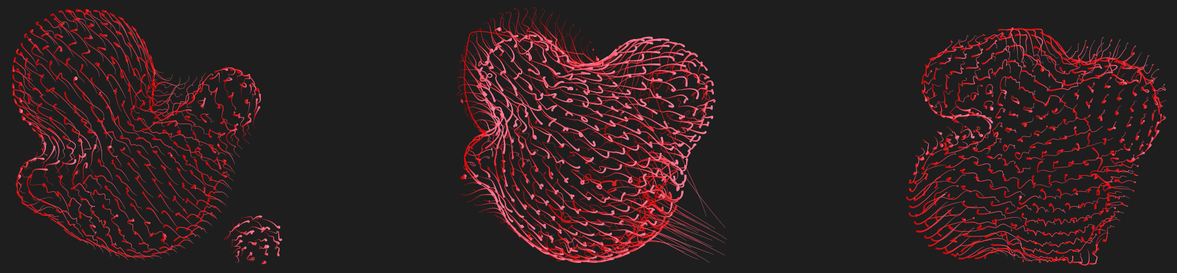


CORRESPONDING ASSEMBLY FORM

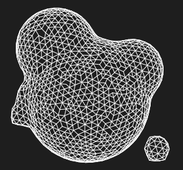
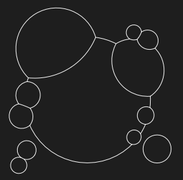
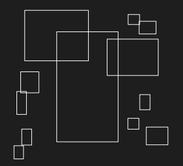
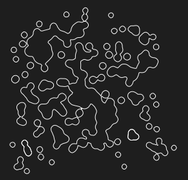
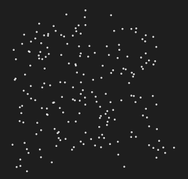
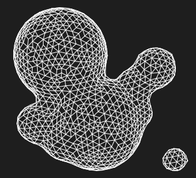
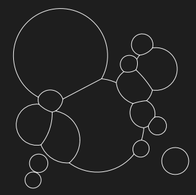
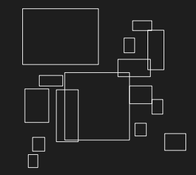
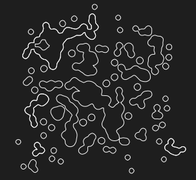
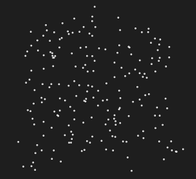
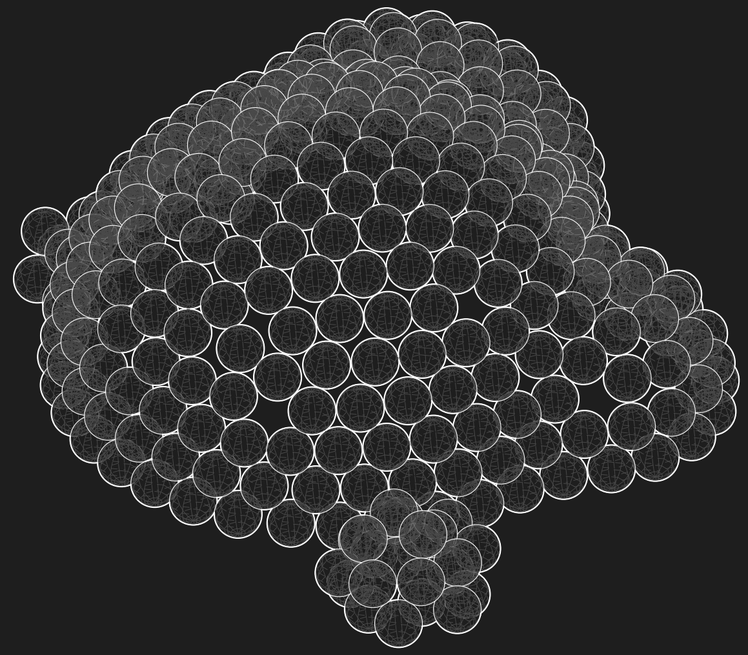


MORPHING TRAILS

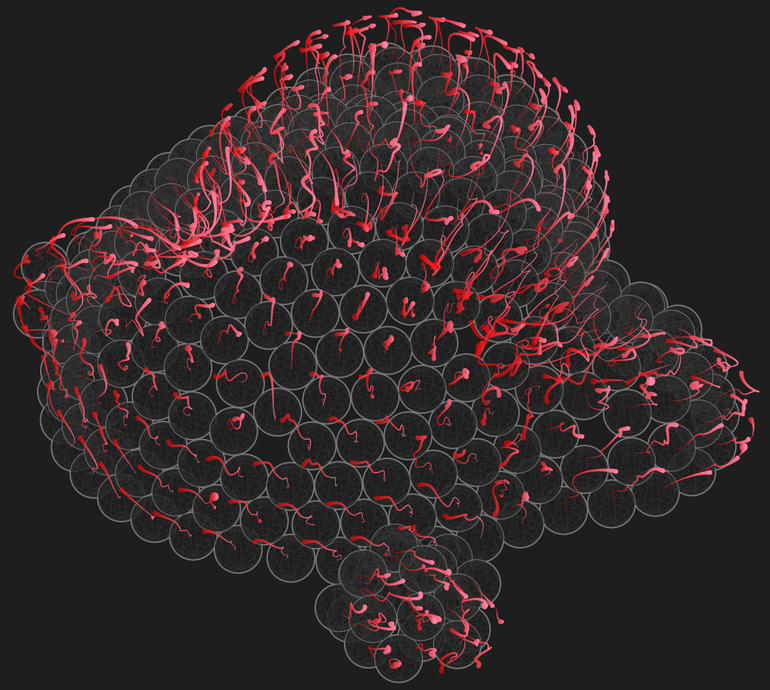




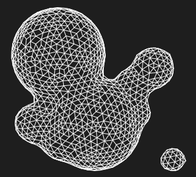
MORPHING TRAILS

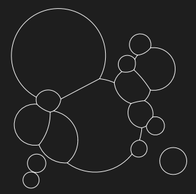












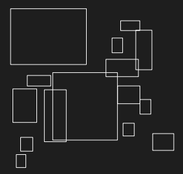
ASSEMBLY FROM SNAPSHOTS #1

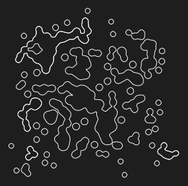


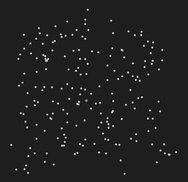
TRAILS FROM SNAPSHOTS #1 TO #2

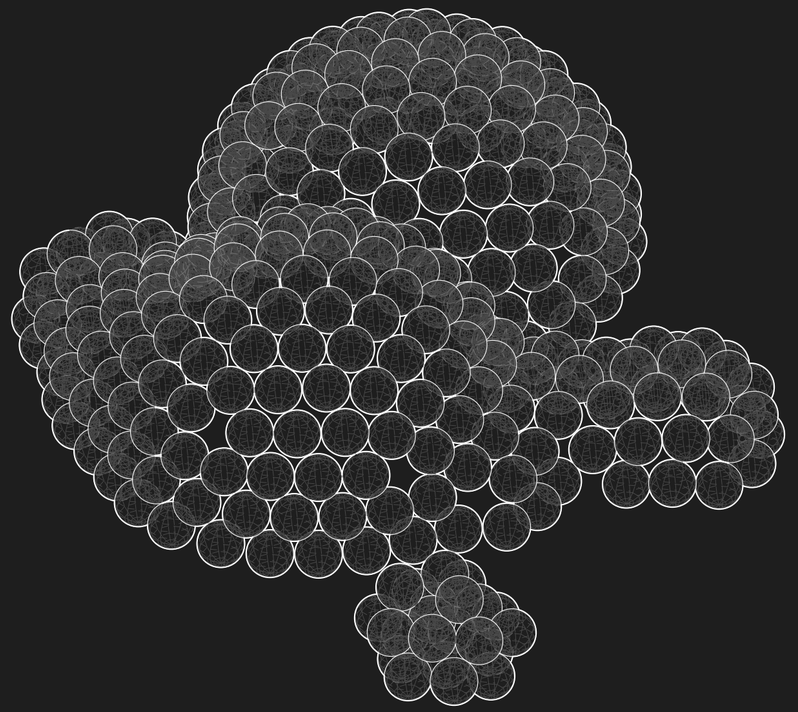




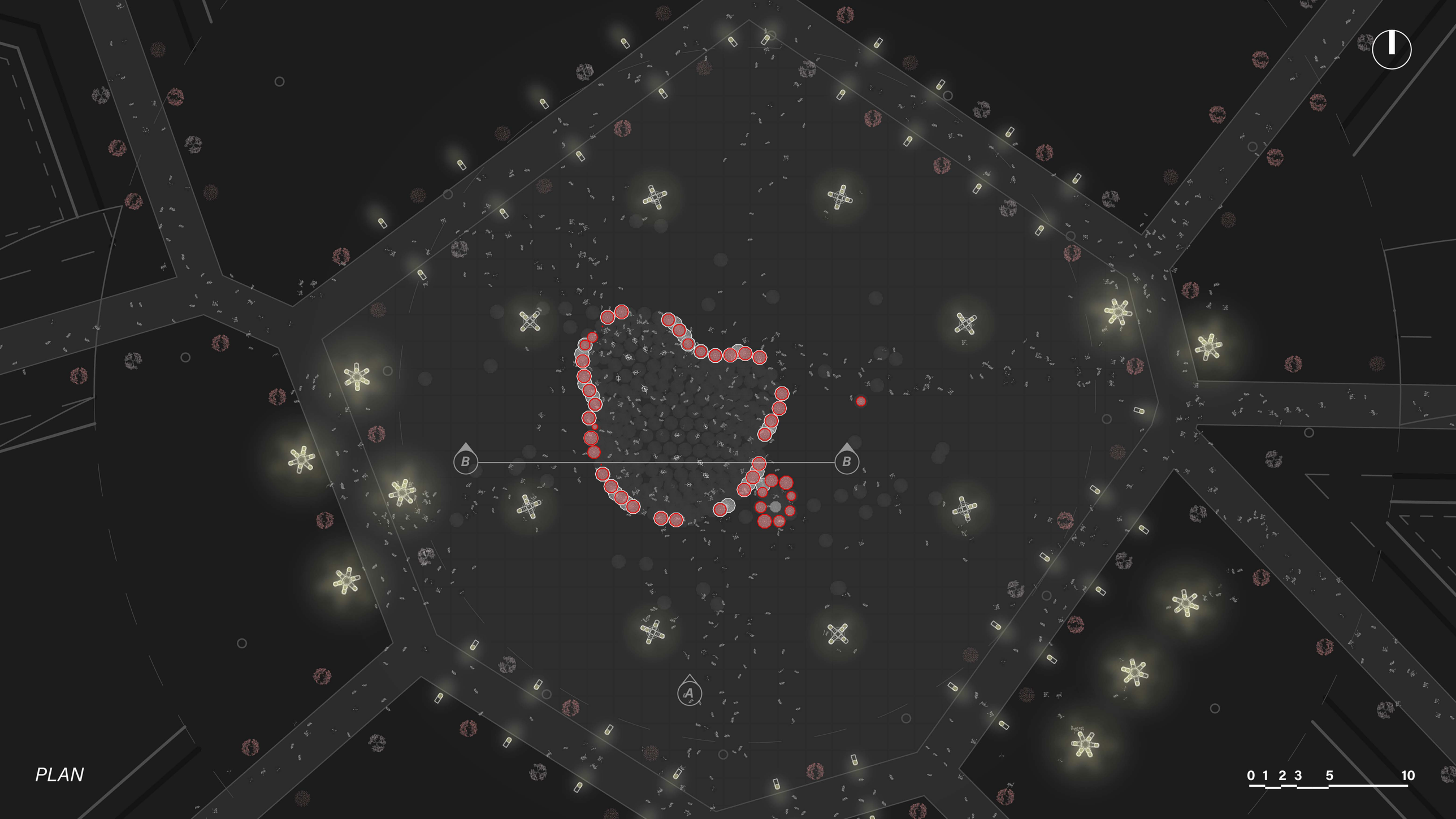




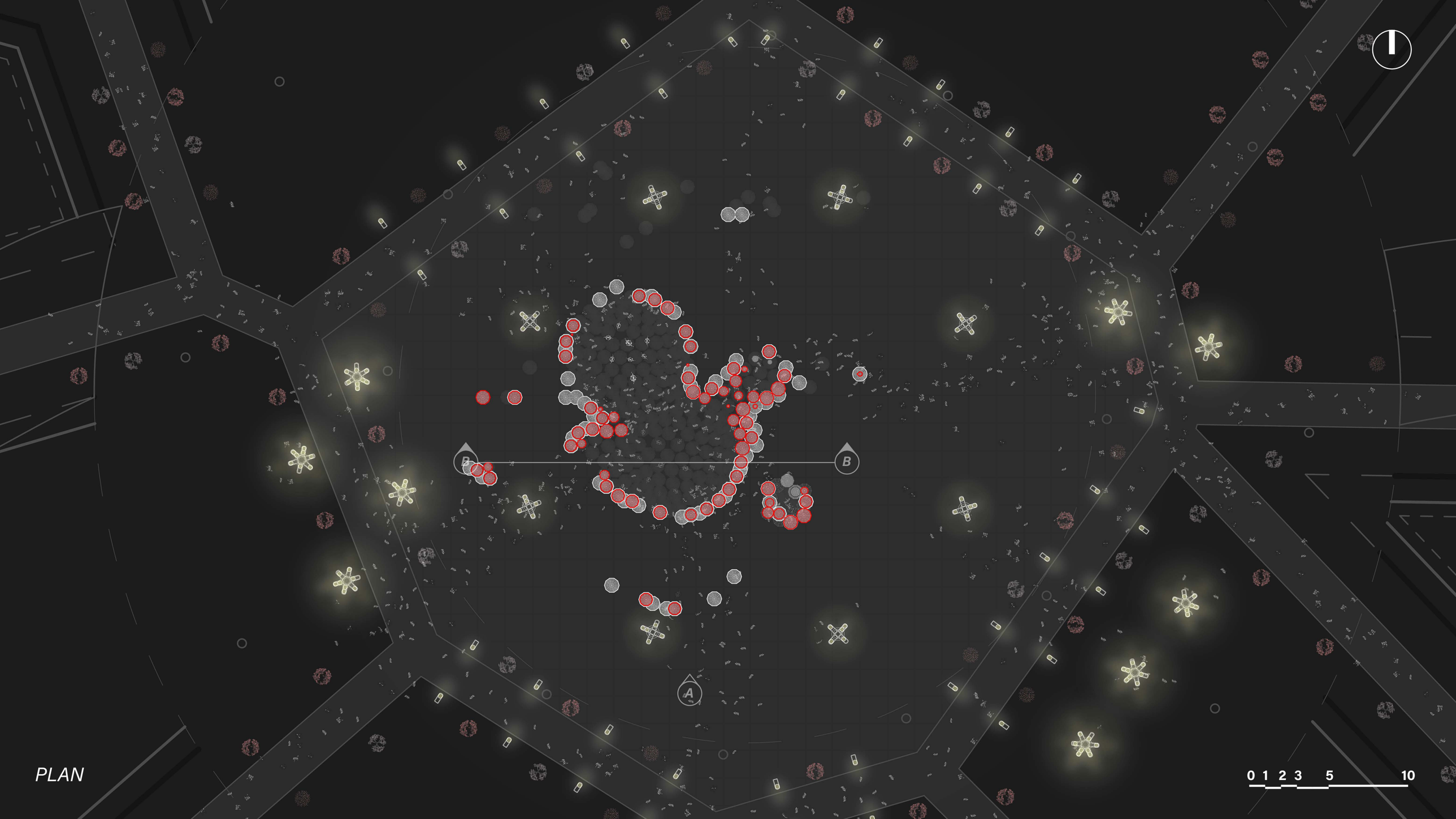




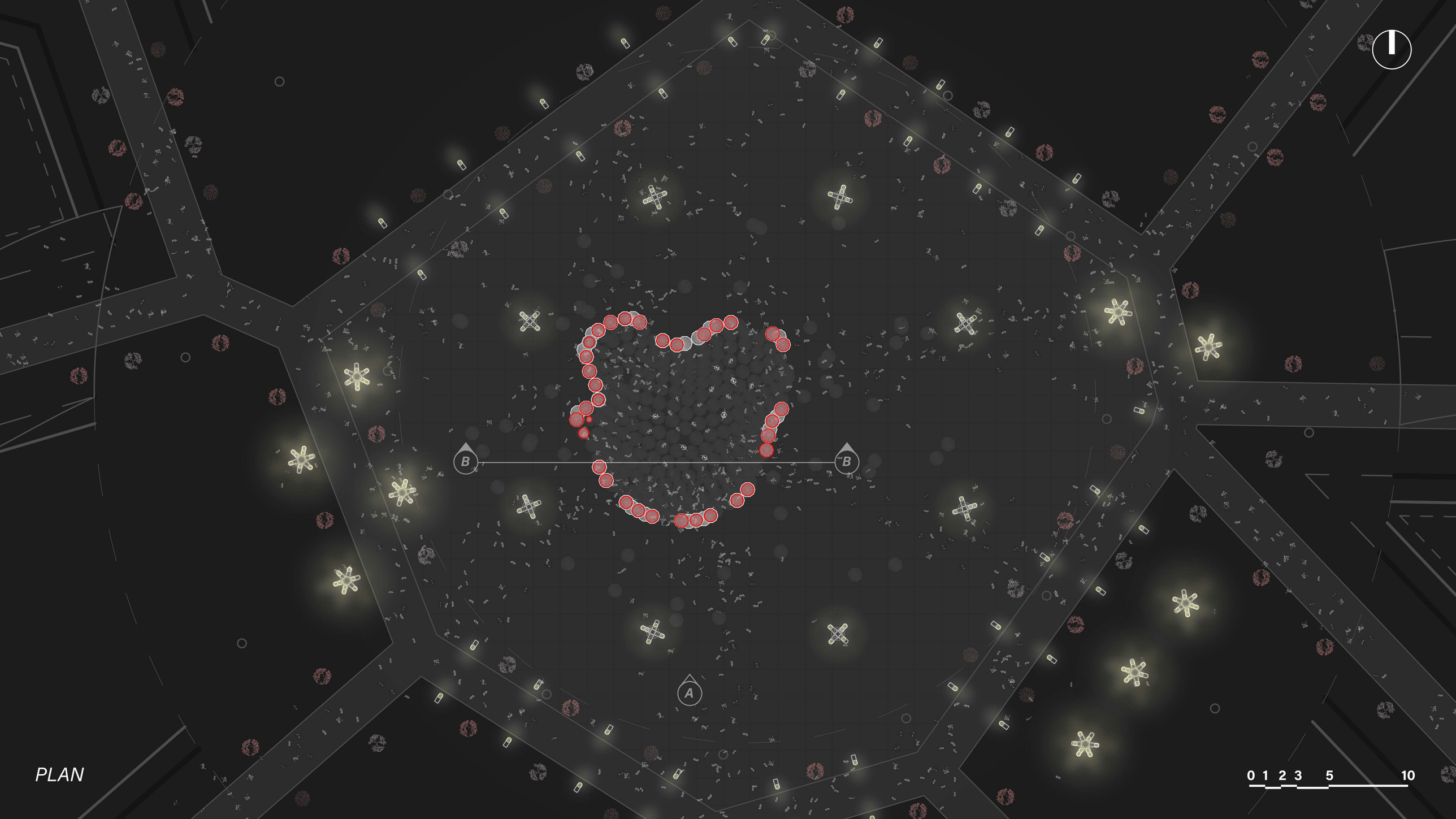
ASSEMBLY FROM SNAPSHOTS #2



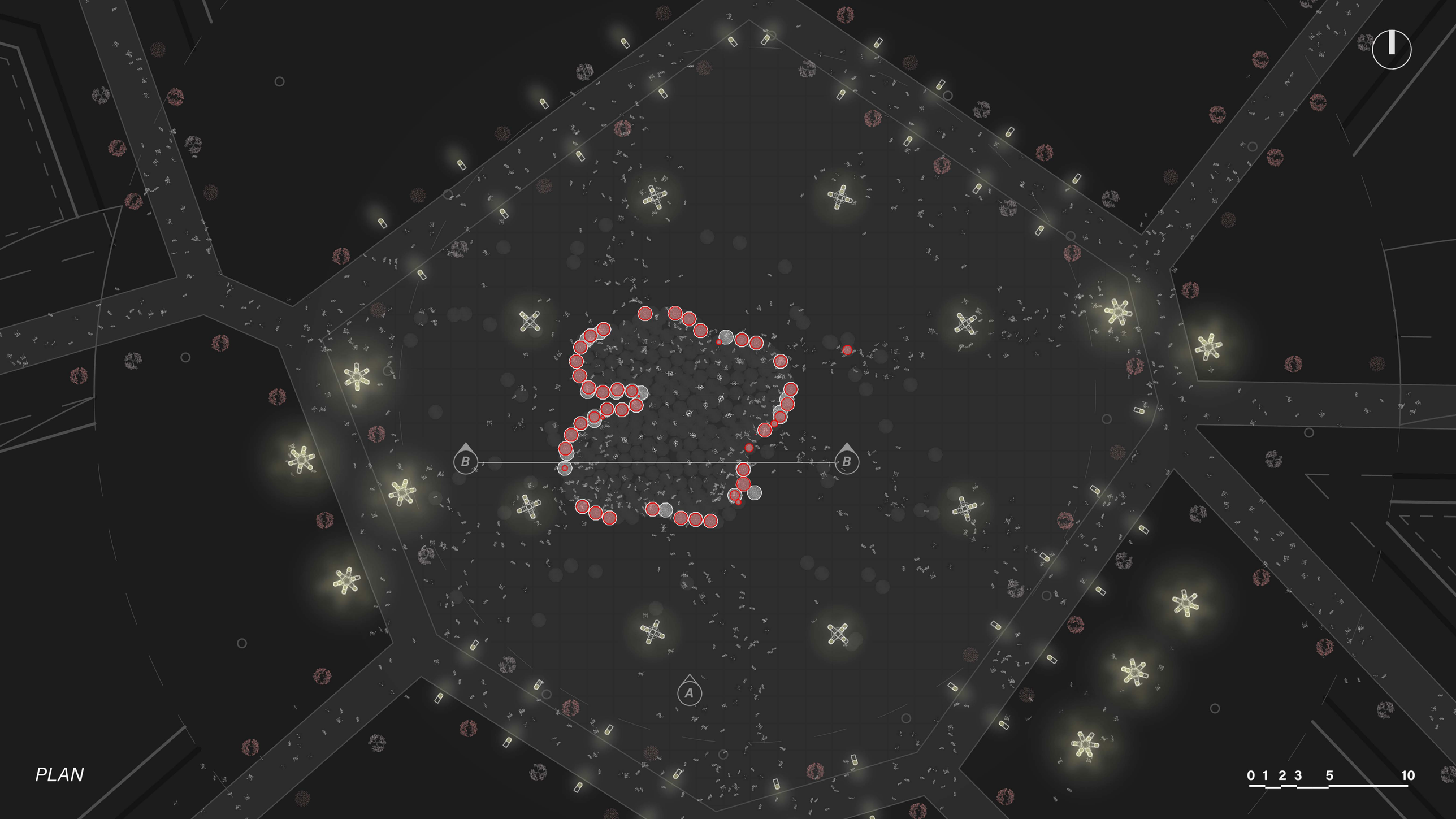
PLAN



PLAN



PLAN



PLAN



ELEVATION

0 1 2 3 5 10



ELEVATION

0 1 2 3 5 10



ELEVATION

0 1 2 3 5 10



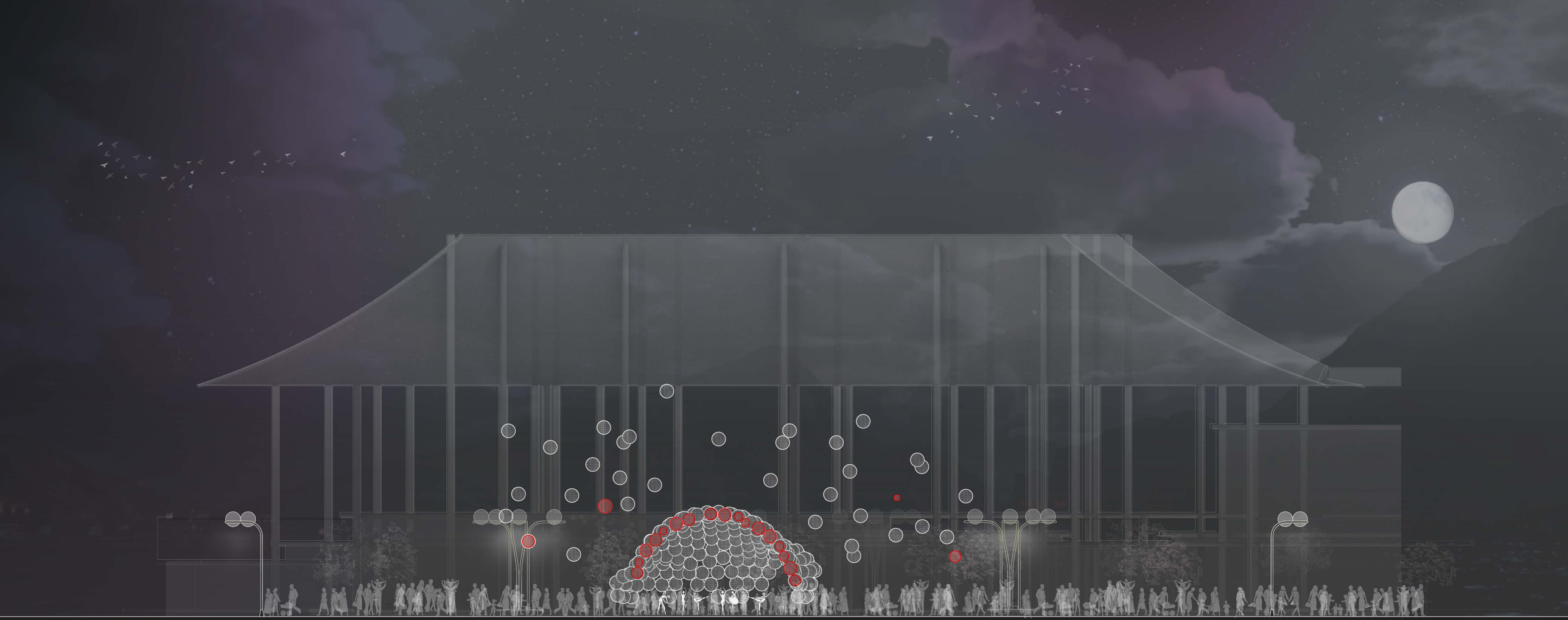
SECTION

0 1 2 3 5 10



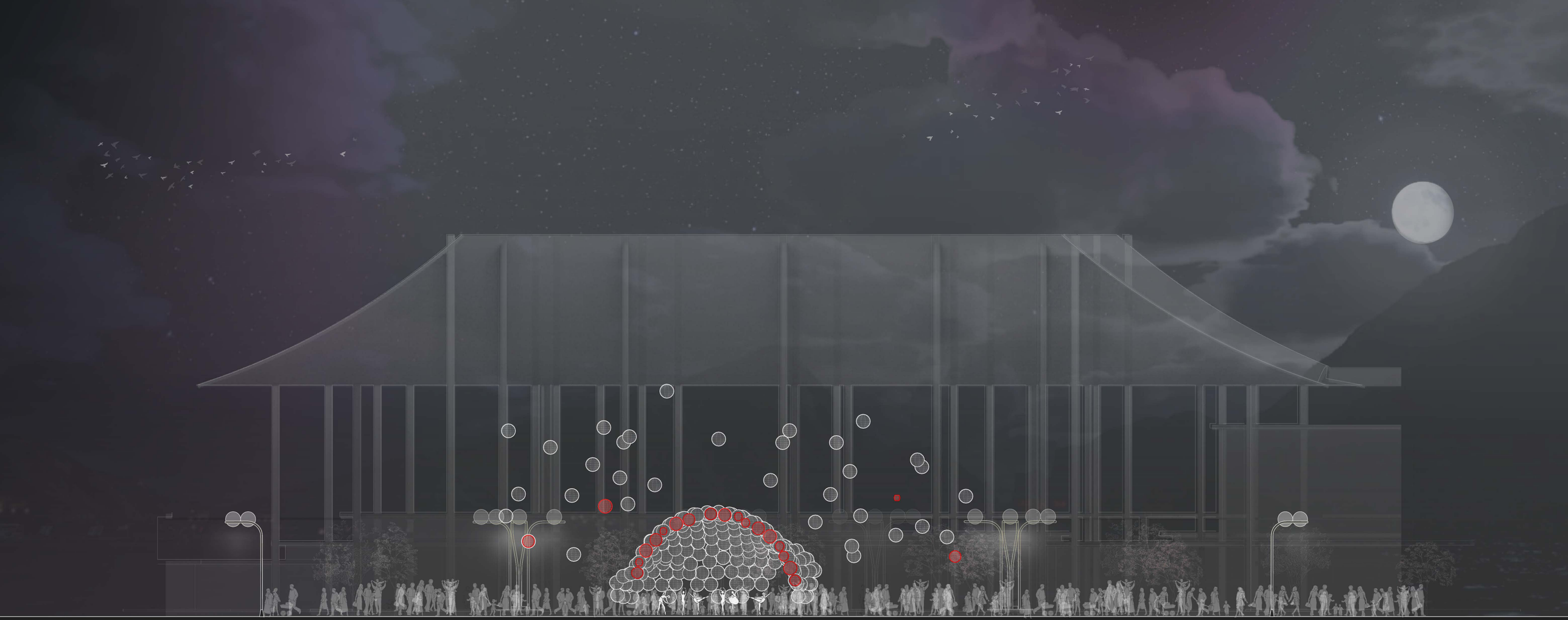
SECTION

0 1 2 3 5 10



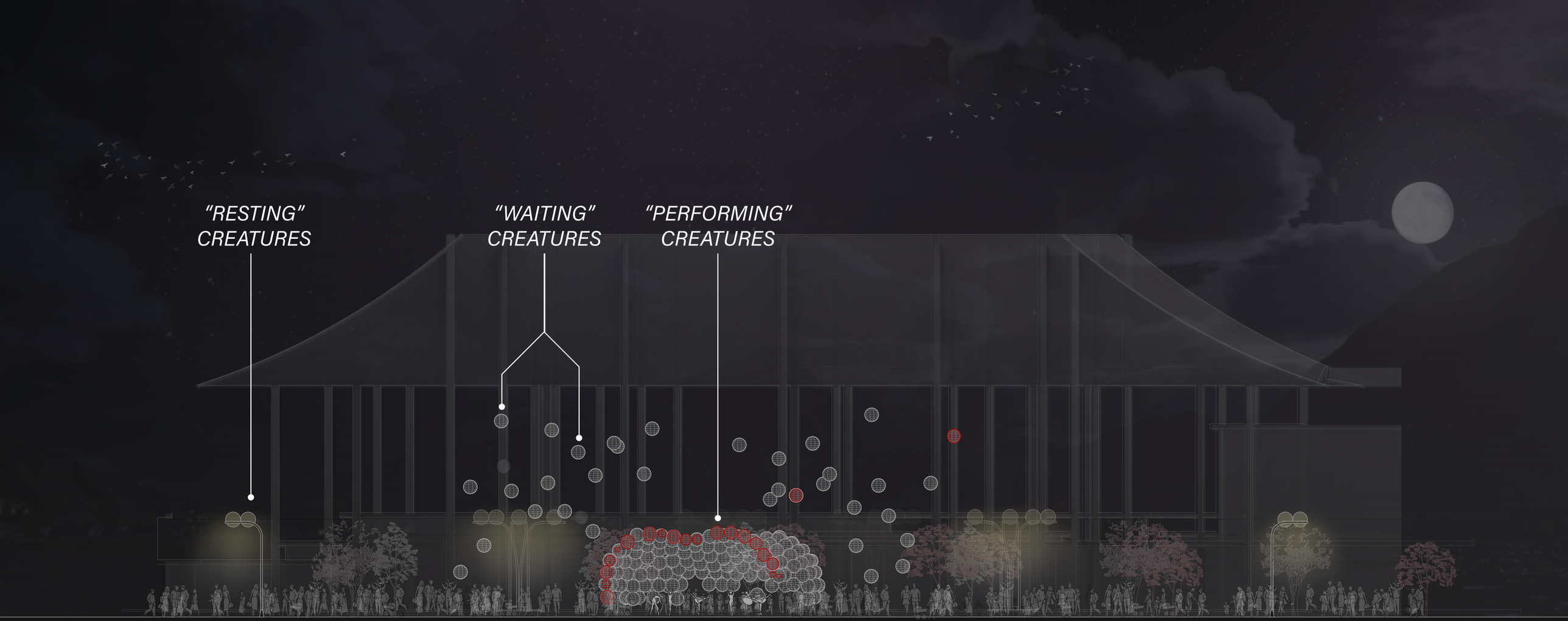
SECTION

0 1 2 3 5 10



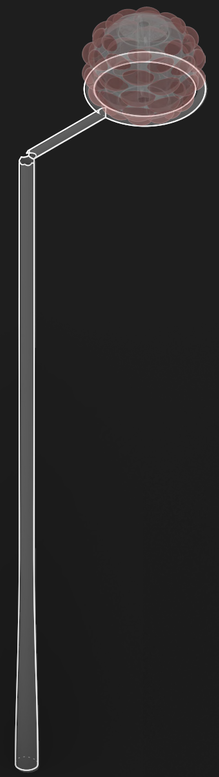
SECTION

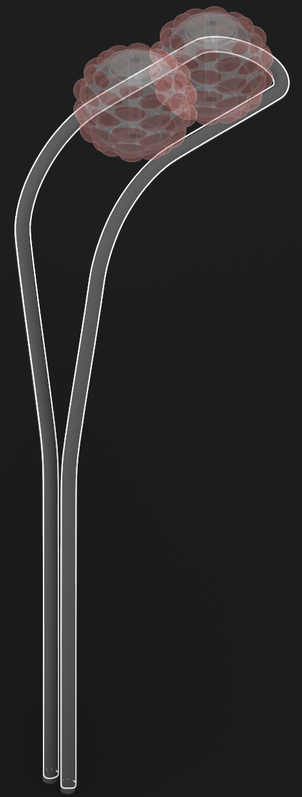
0 1 2 3 5 10

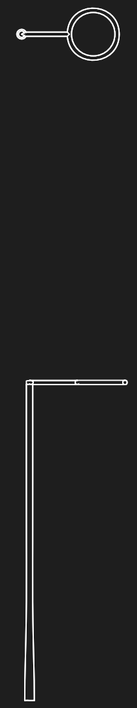


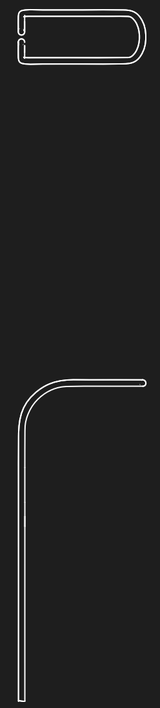
CONTINUOUS CREATURE CHAIN

0 1 2 3 5 10

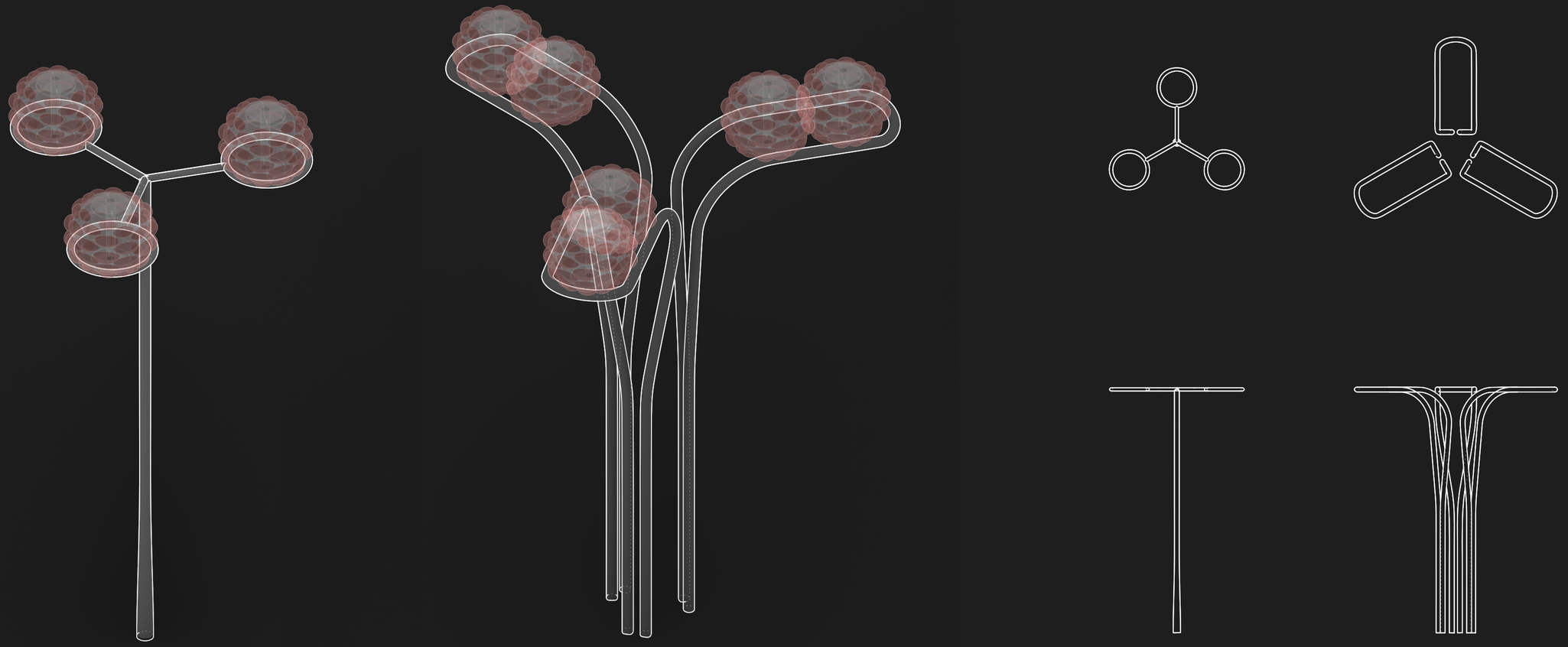




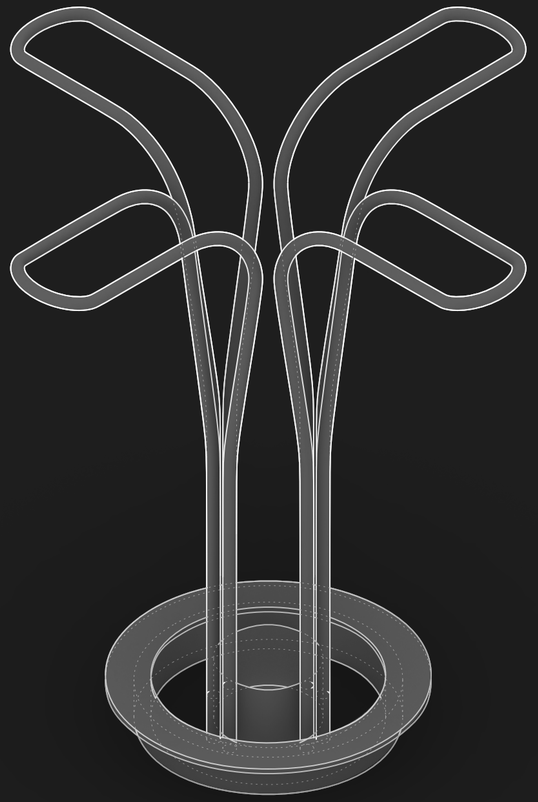


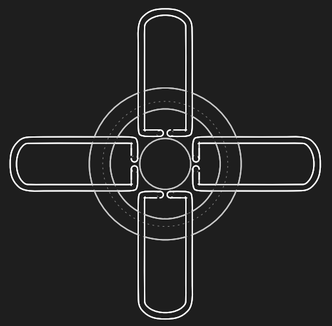


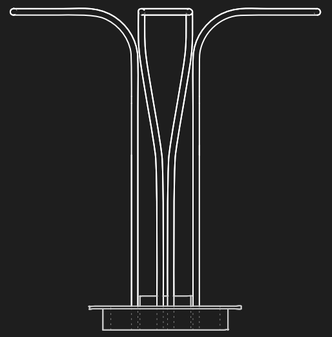
INTERACTIONS: STREET LIGHTING / URBAN FURNITURE



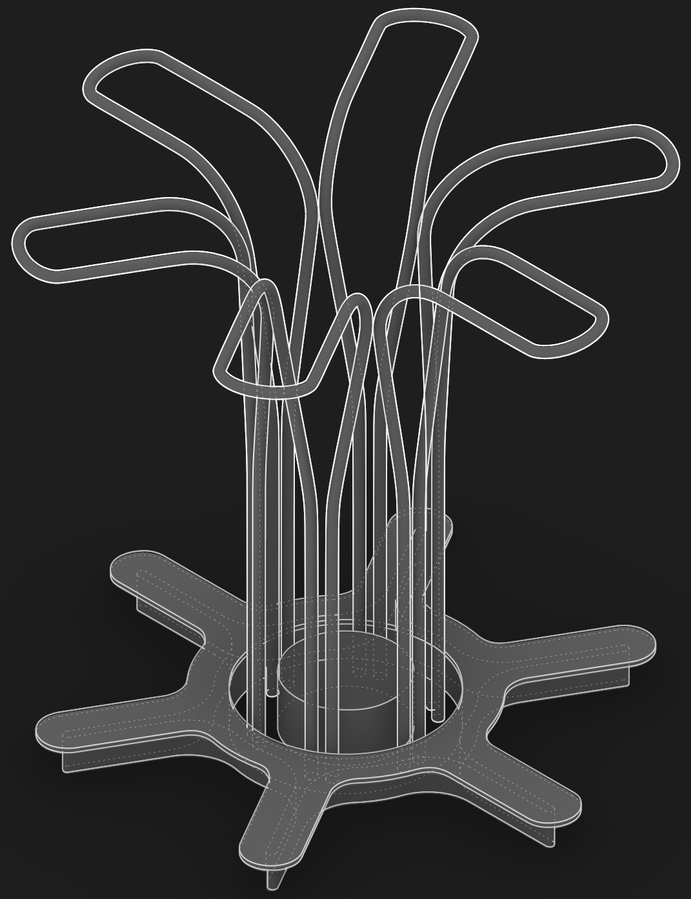
INTERACTIONS: STREET LIGHTING / URBAN FURNITURE

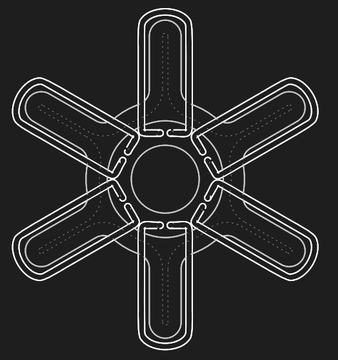


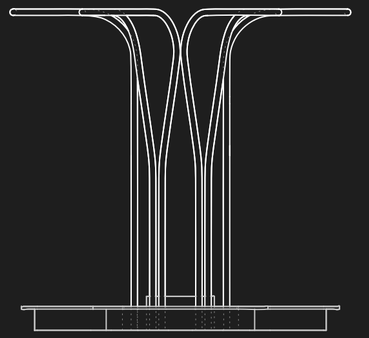




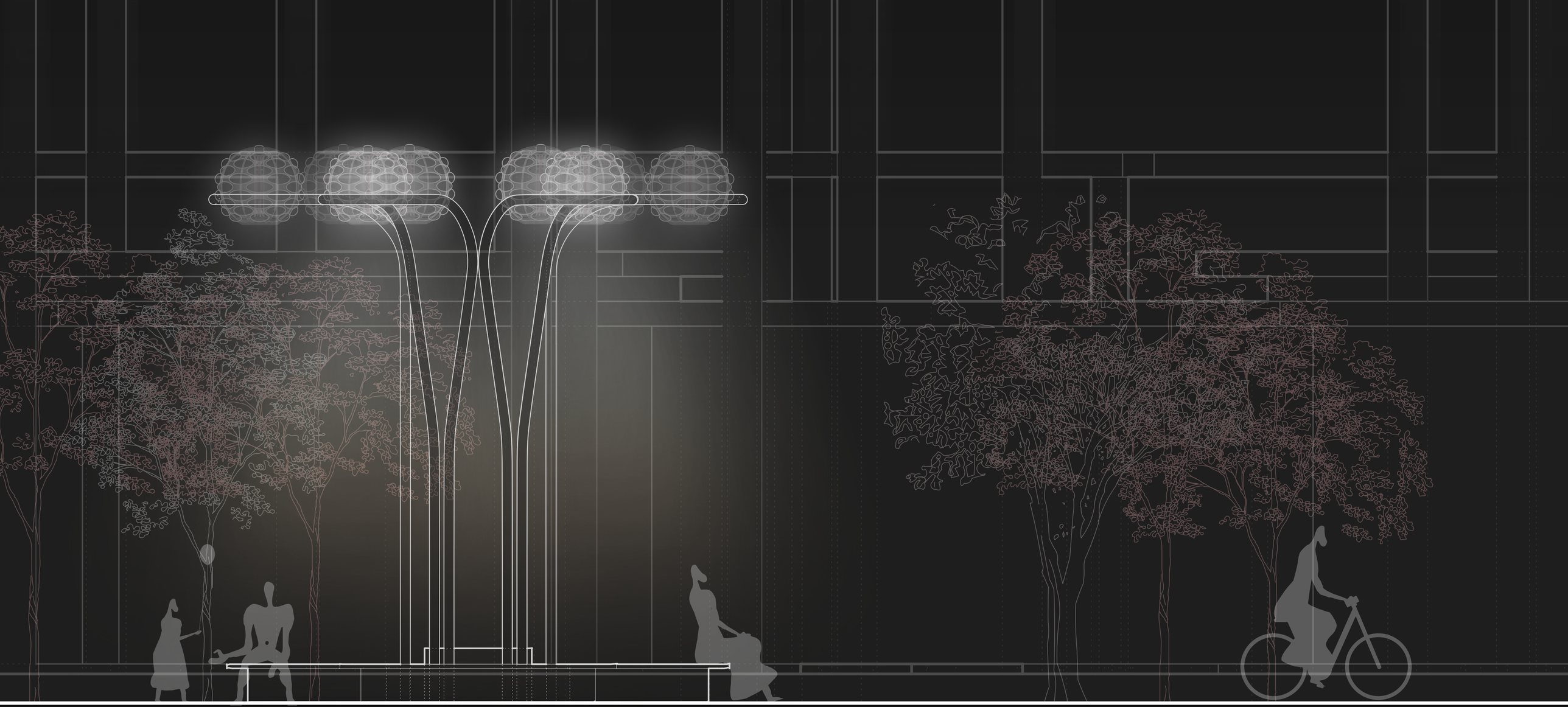
INTERACTIONS: STREET LIGHTING / URBAN FURNITURE



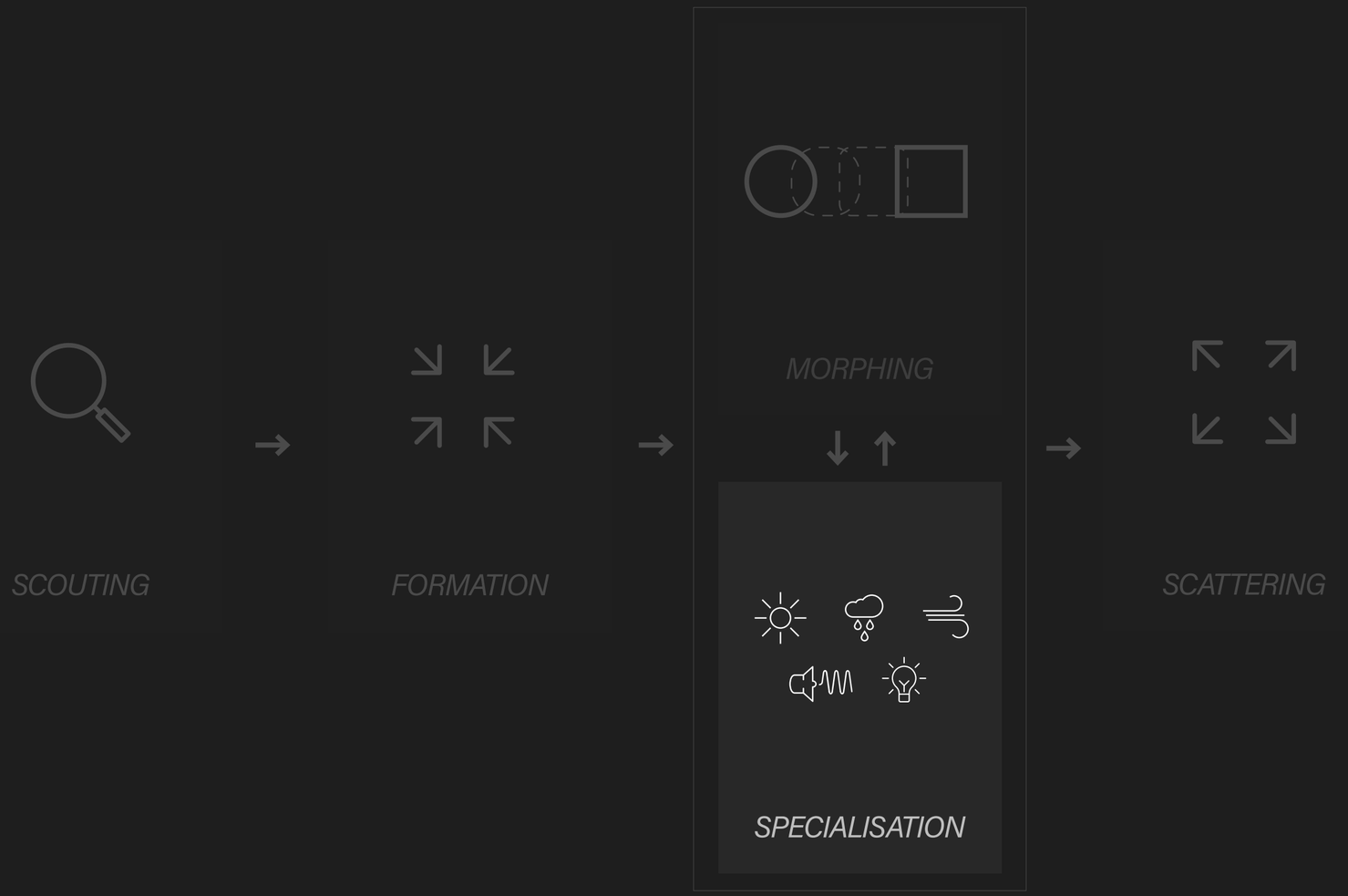




INTERACTIONS: STREET LIGHTING / URBAN FURNITURE



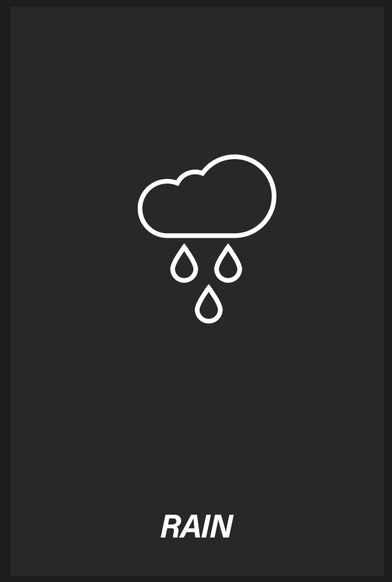
INTERACTIONS: STREET LIGHTING / URBAN FURNITURE



ASSEMBLY: SPECIALISATION PROCESS



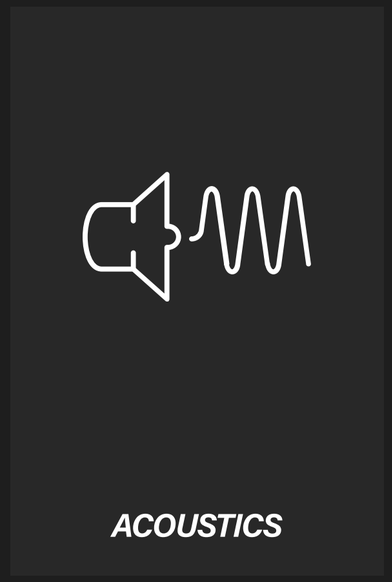
WIND



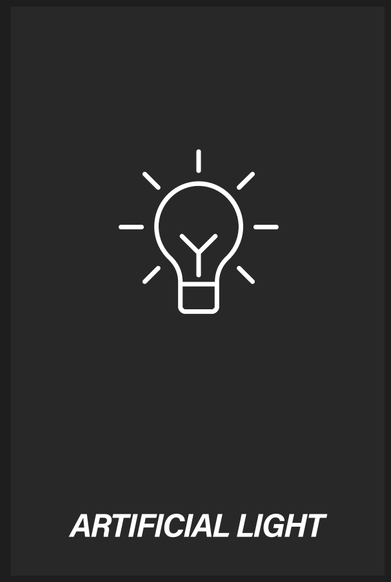
RAIN



SUN



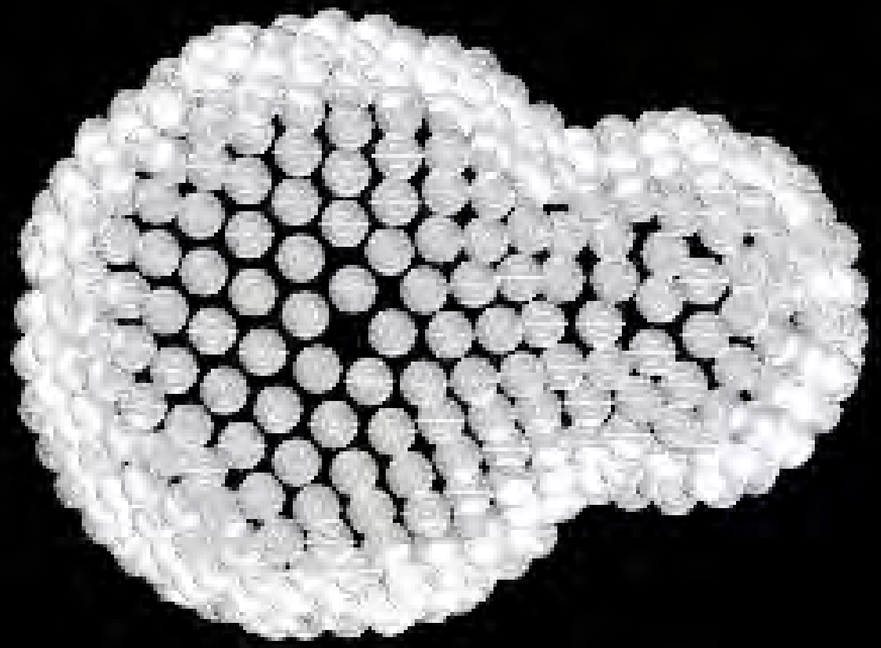
ACOUSTICS

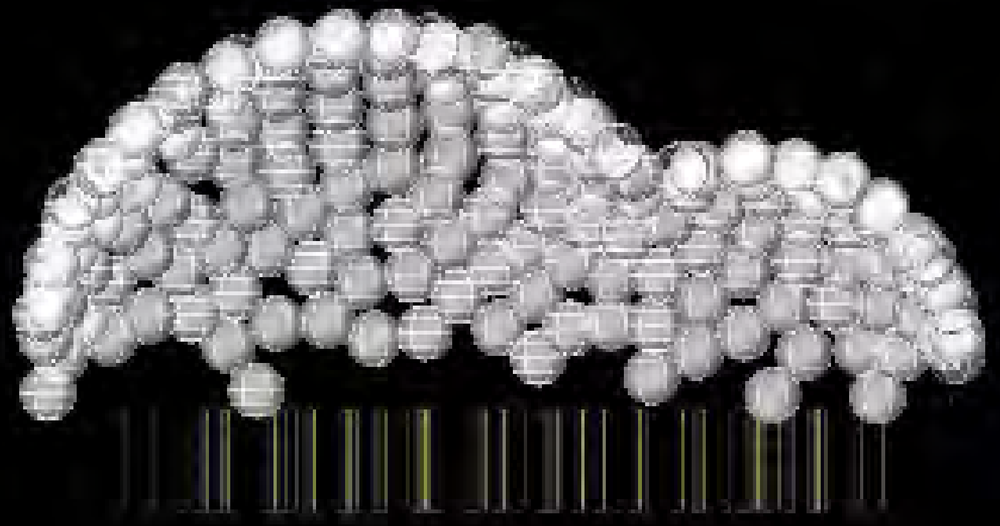


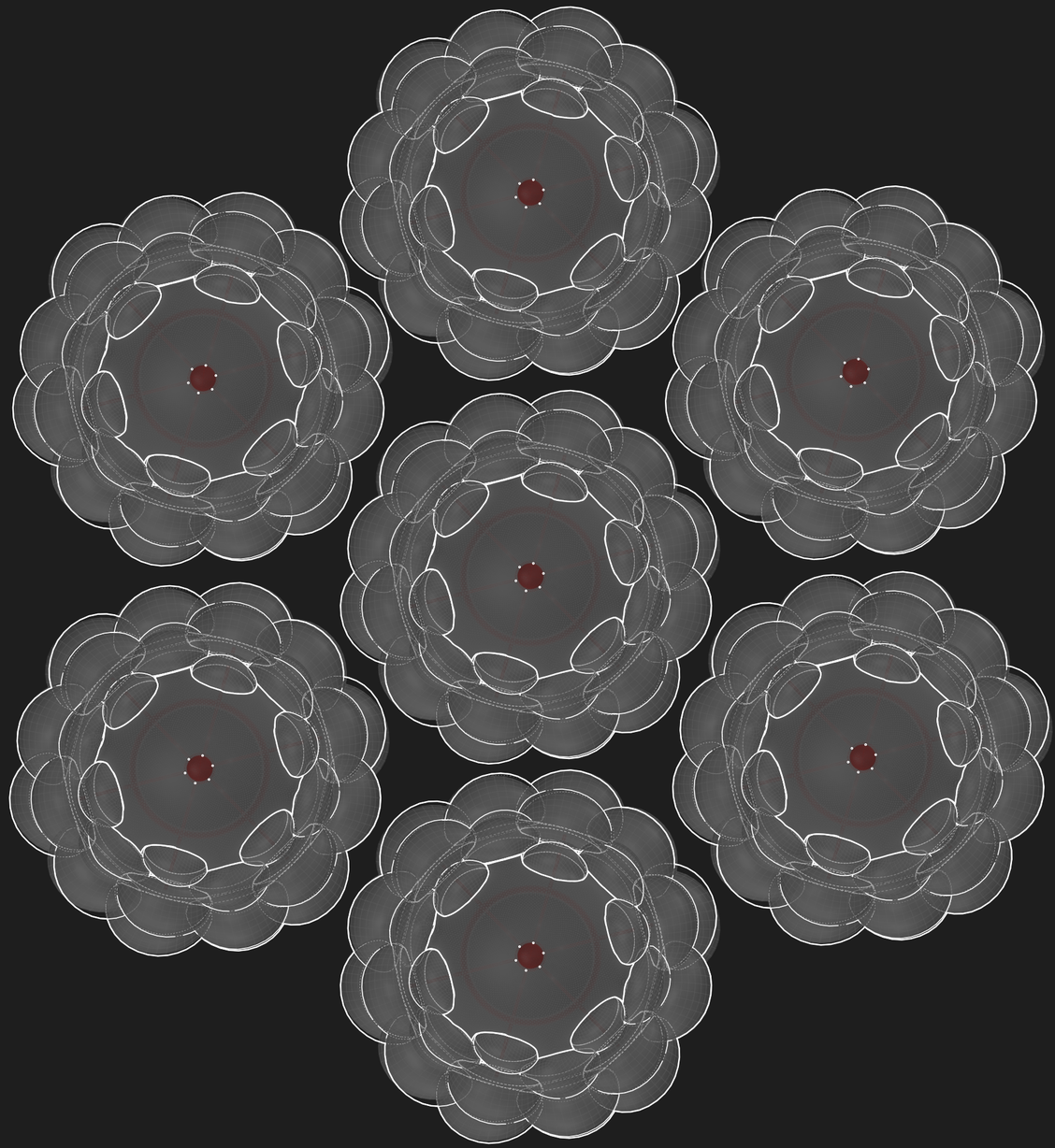
ARTIFICIAL LIGHT

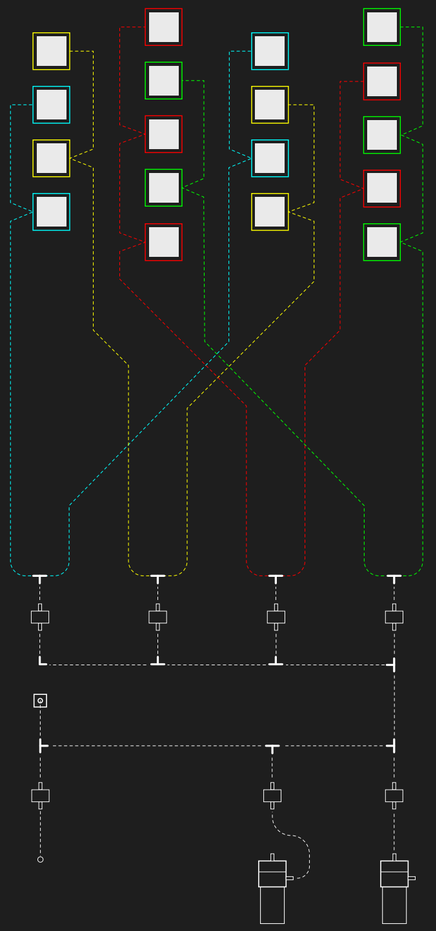
ASSEMBLY: SPECIALISATION PROCESS



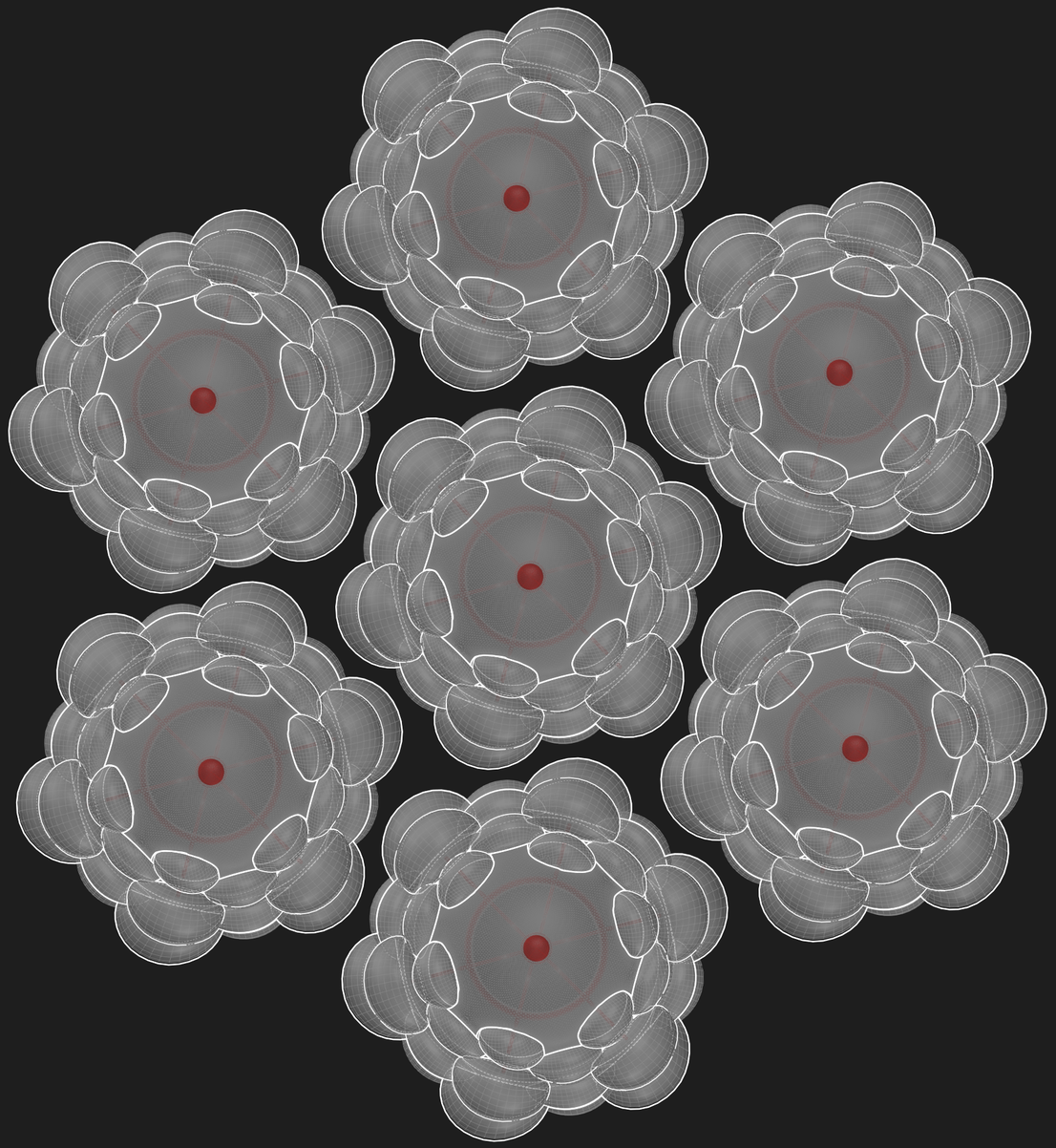


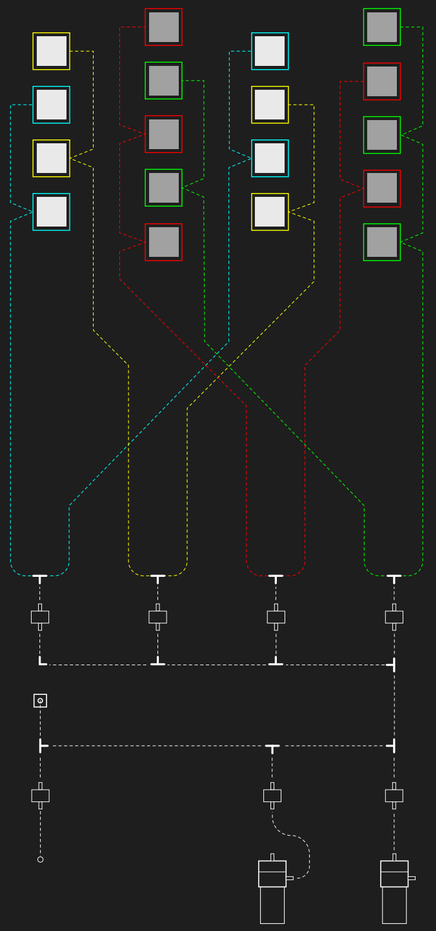




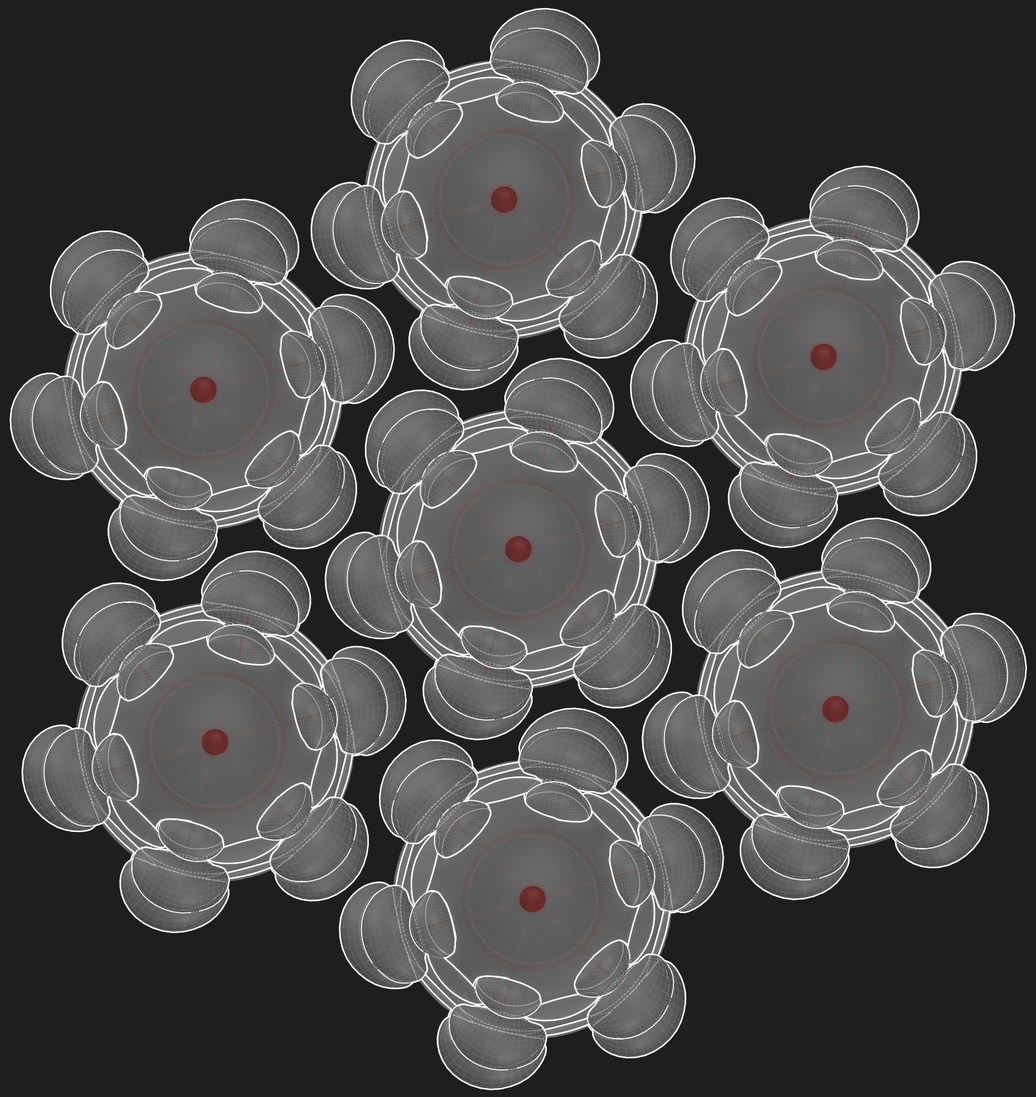


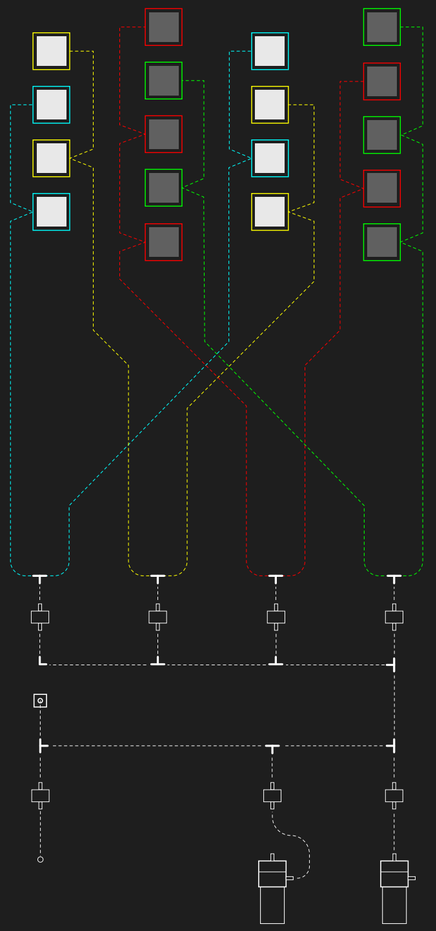
ACTUATION: CONNECTIONS



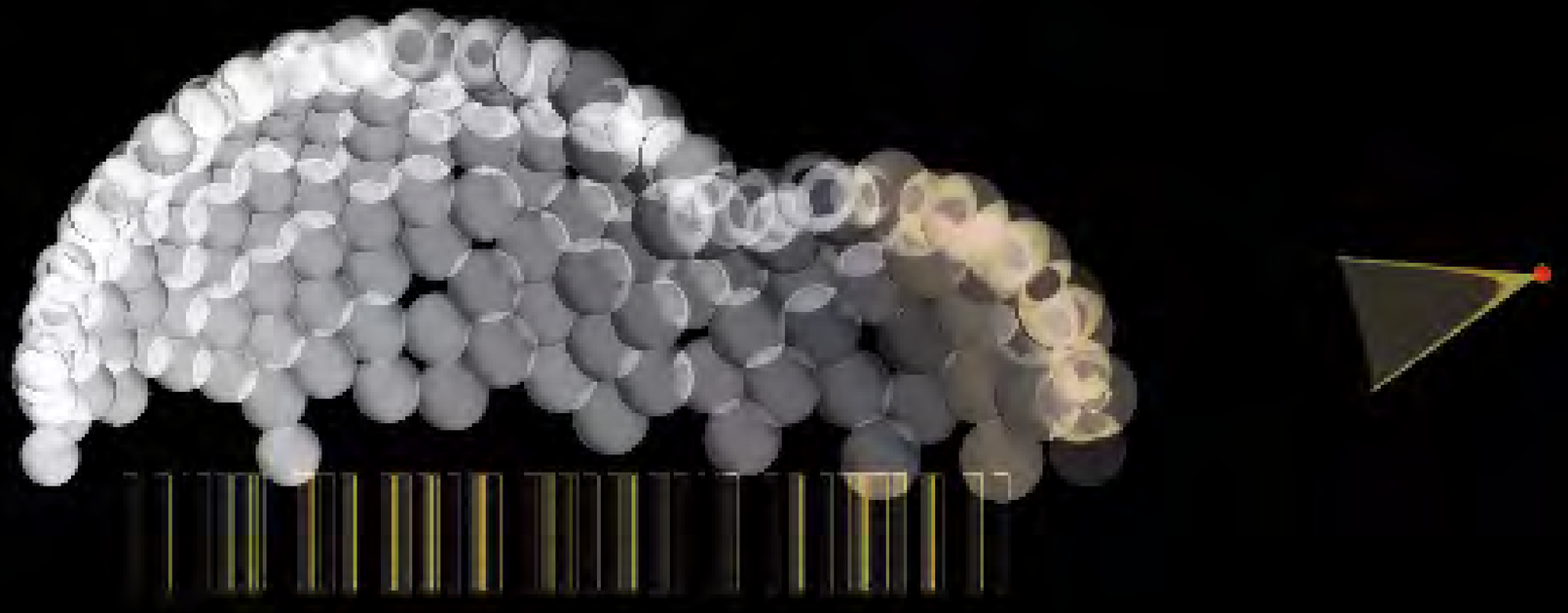


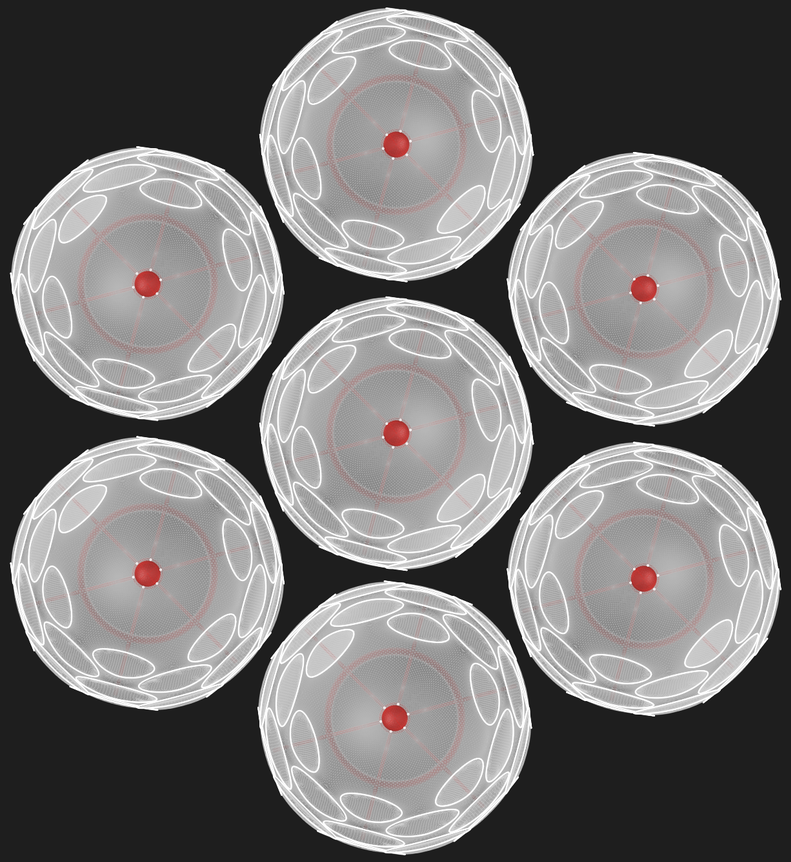
ACTUATION: CONNECTIONS

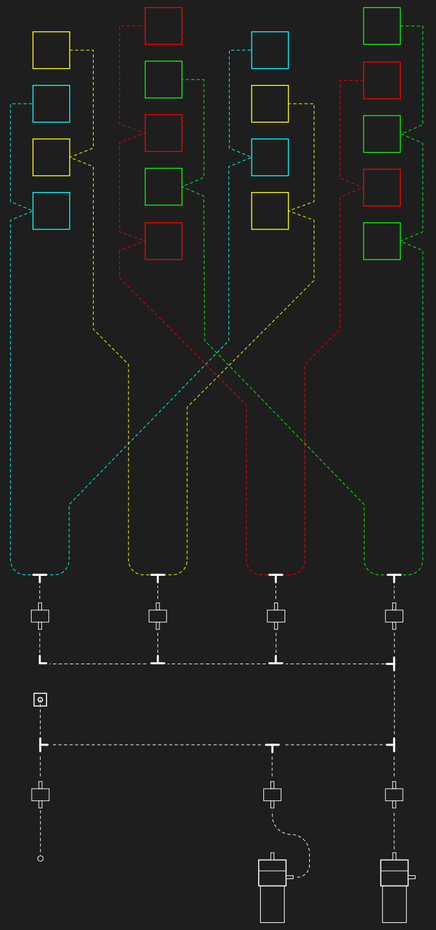




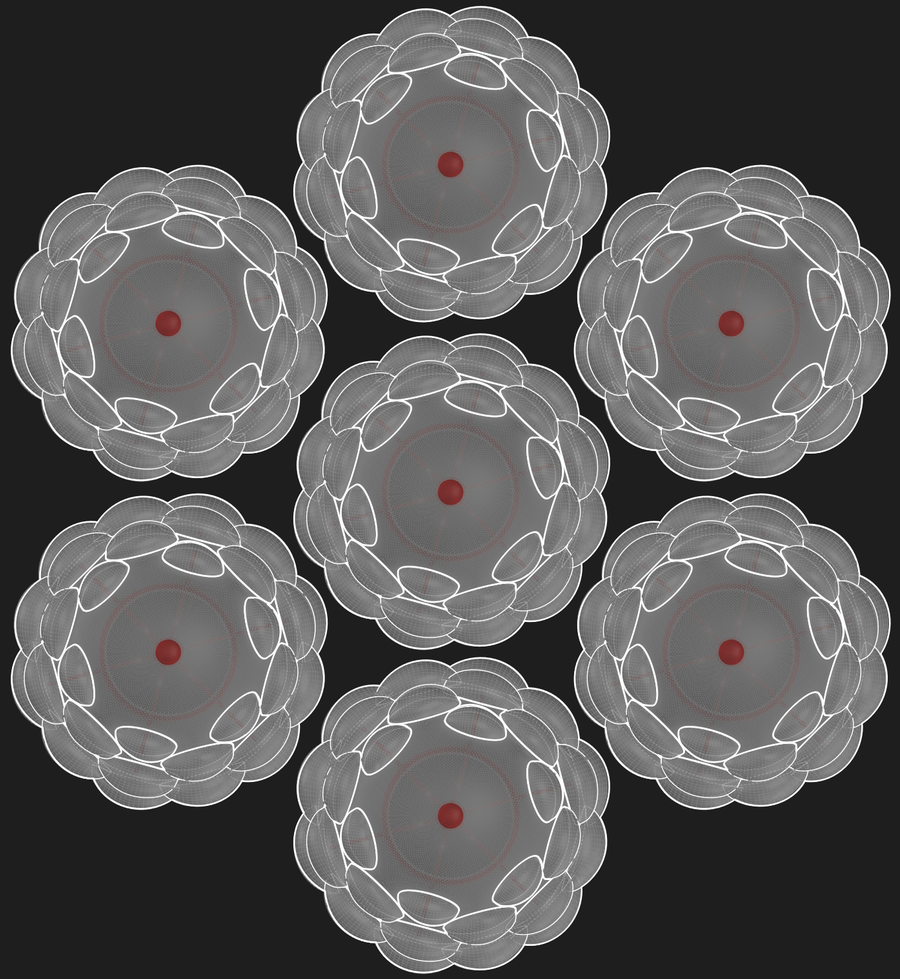
ACTUATION: CONNECTIONS

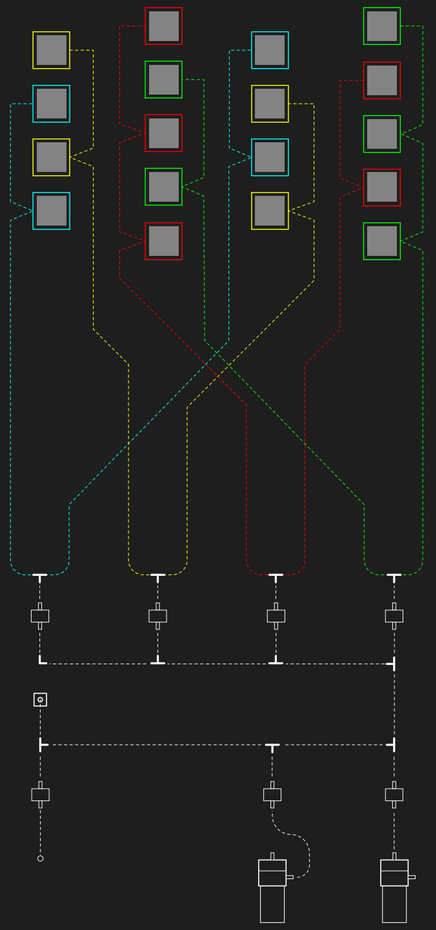




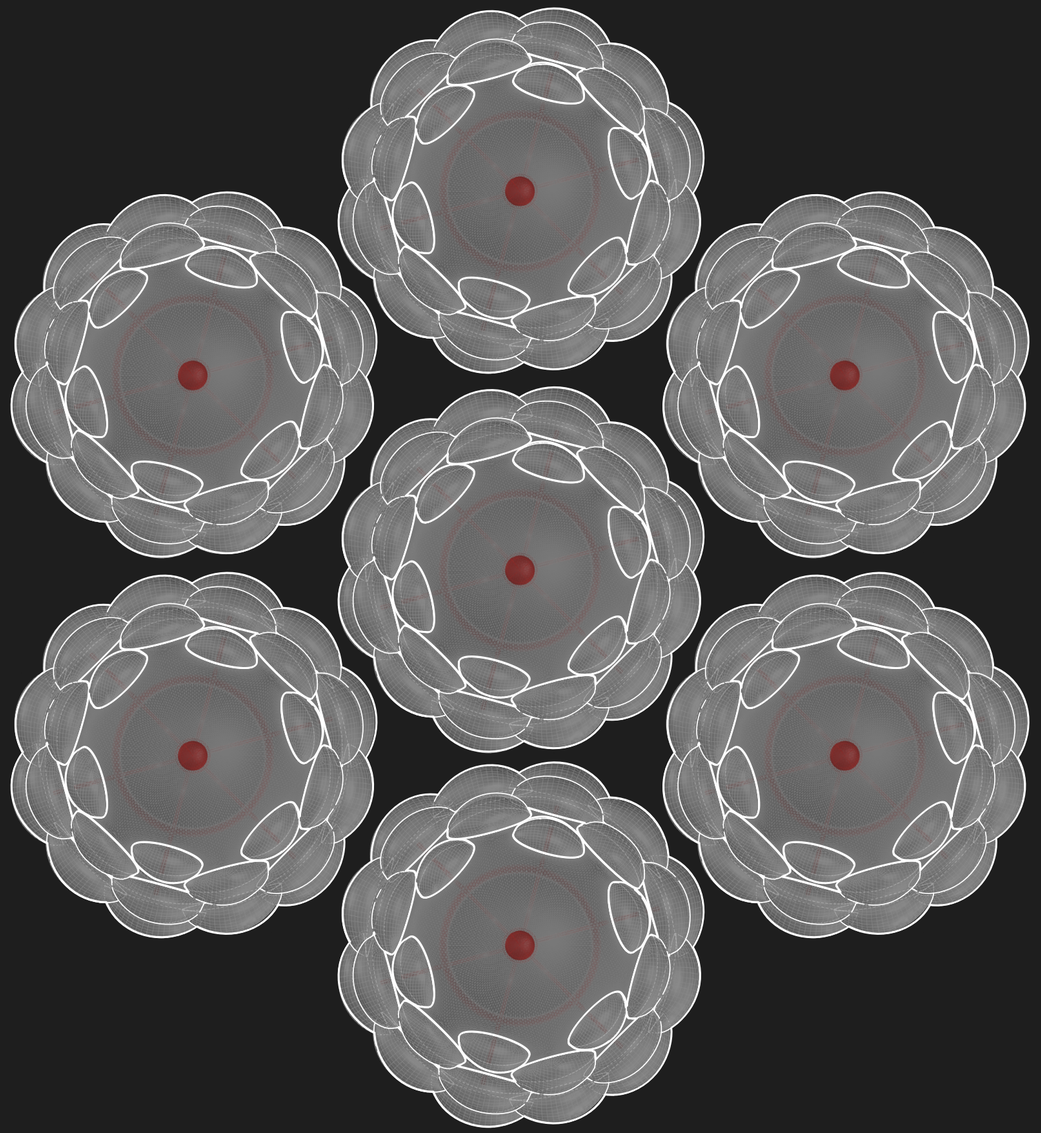


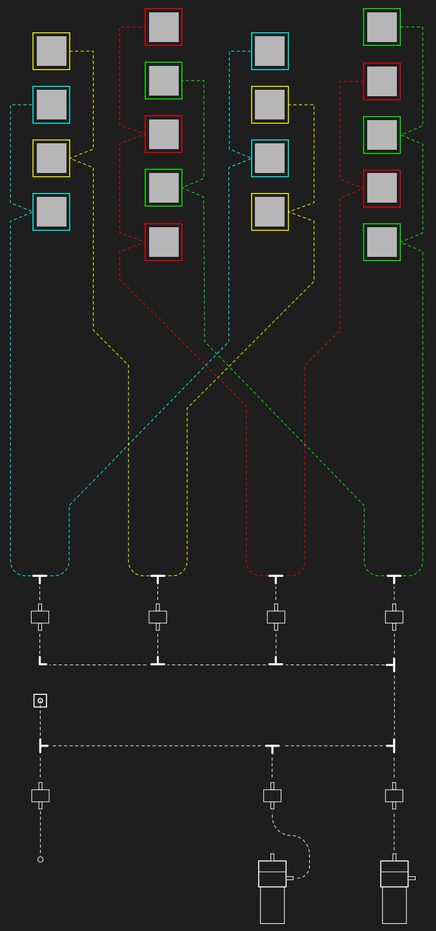
ACTUATION: CONNECTIONS



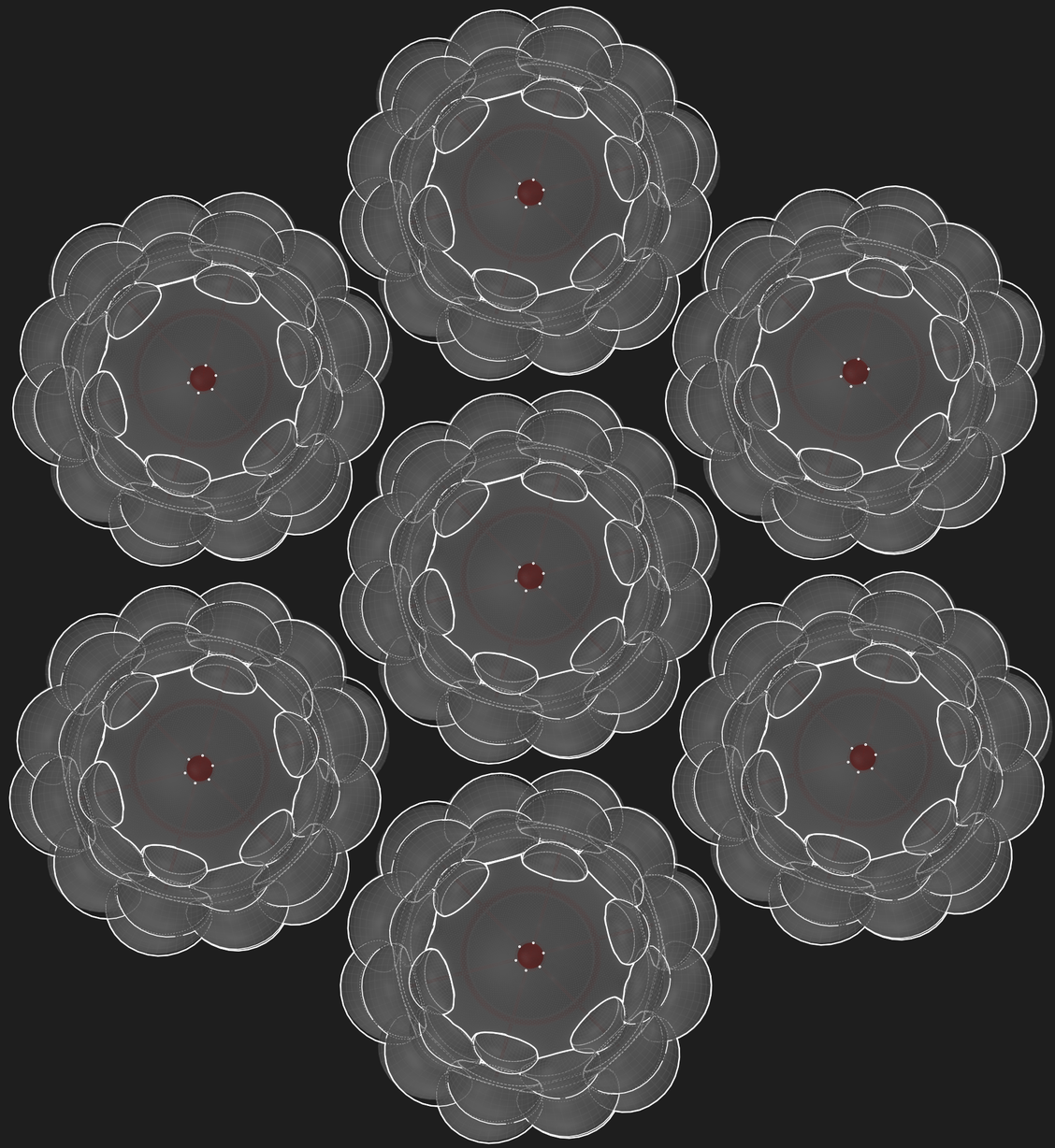


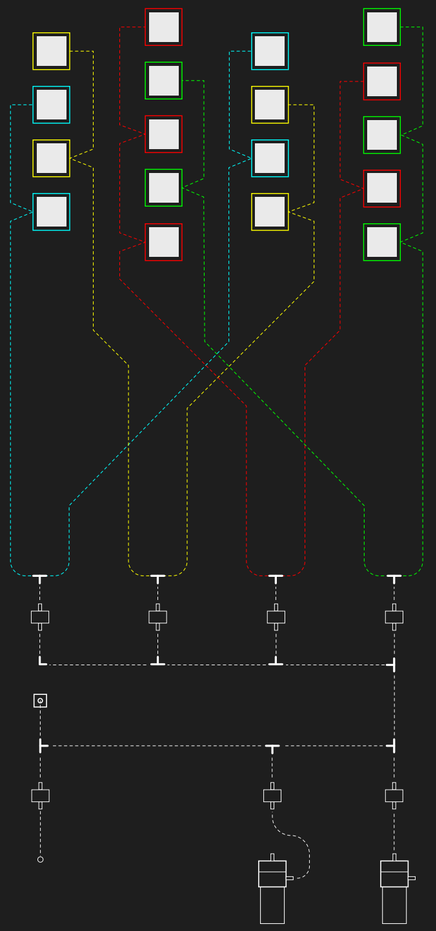
ACTUATION: CONNECTIONS



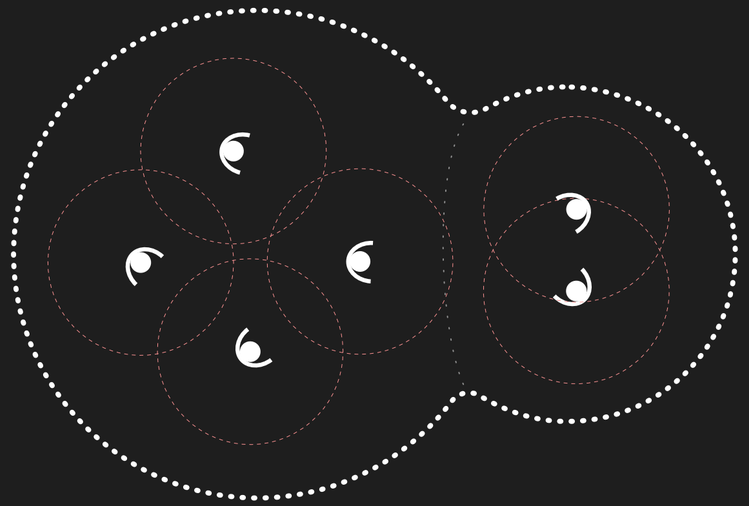


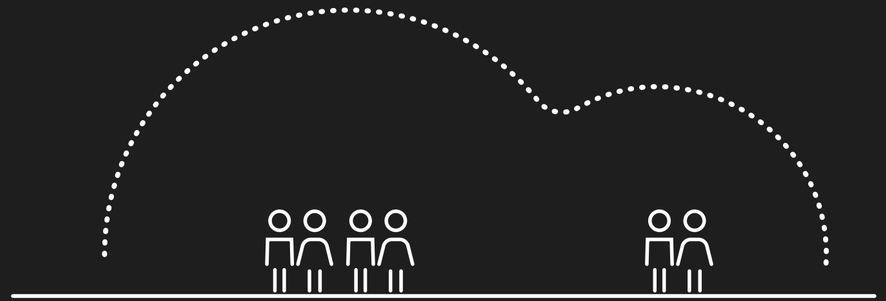
ACTUATION: CONNECTIONS



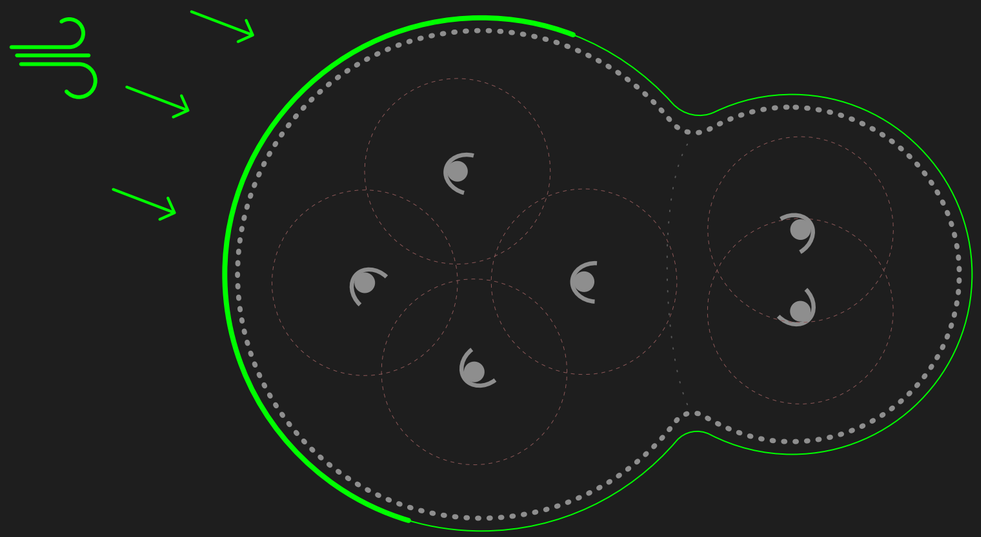


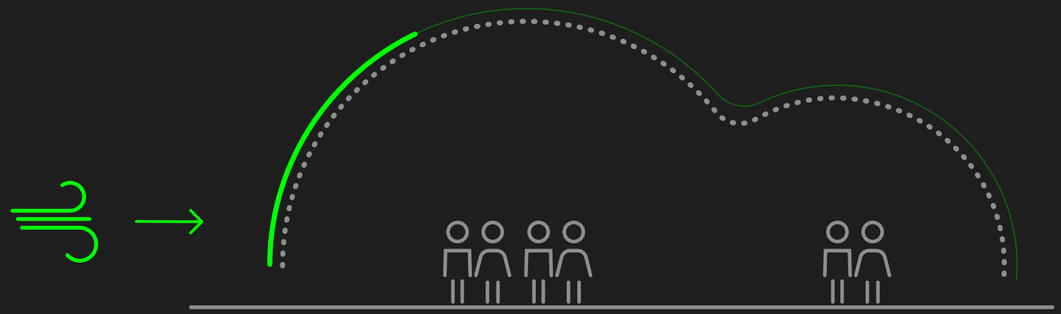
ACTUATION: CONNECTIONS



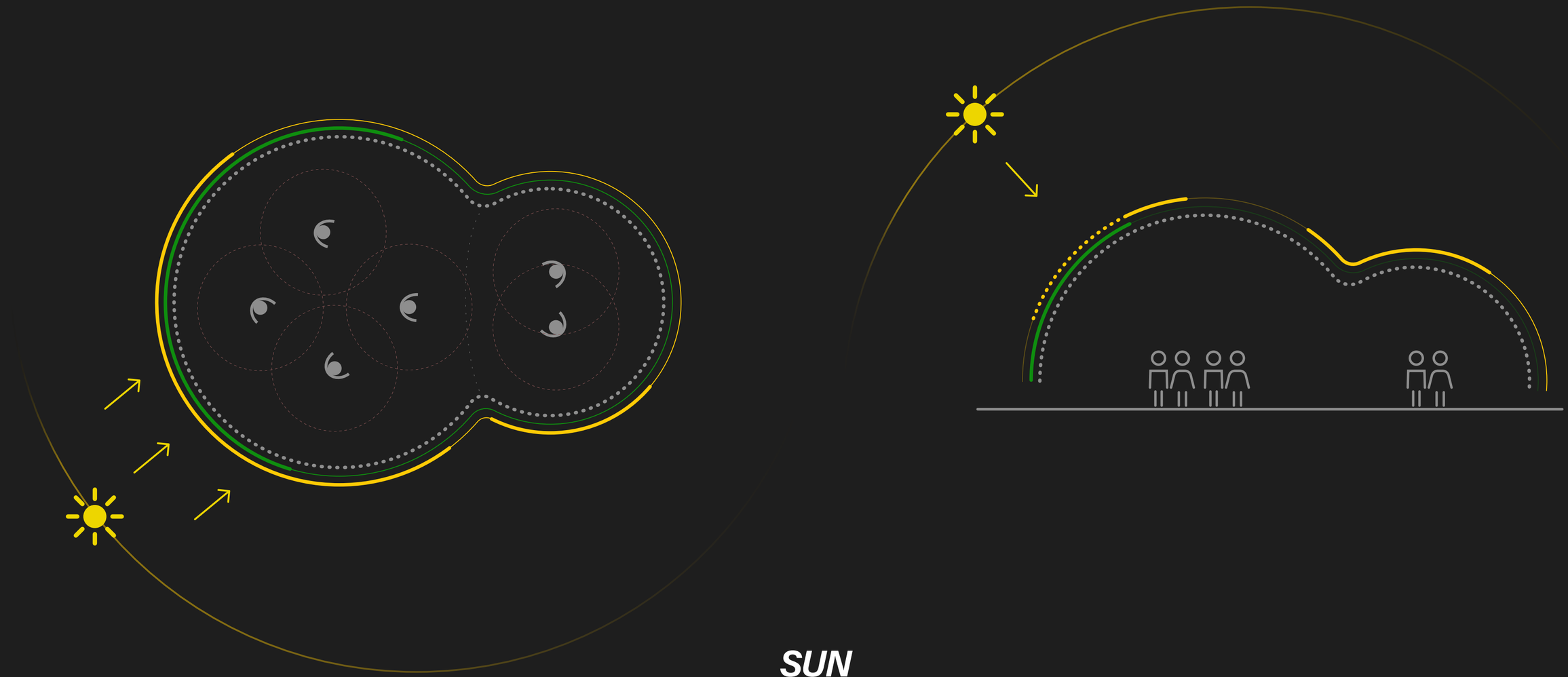


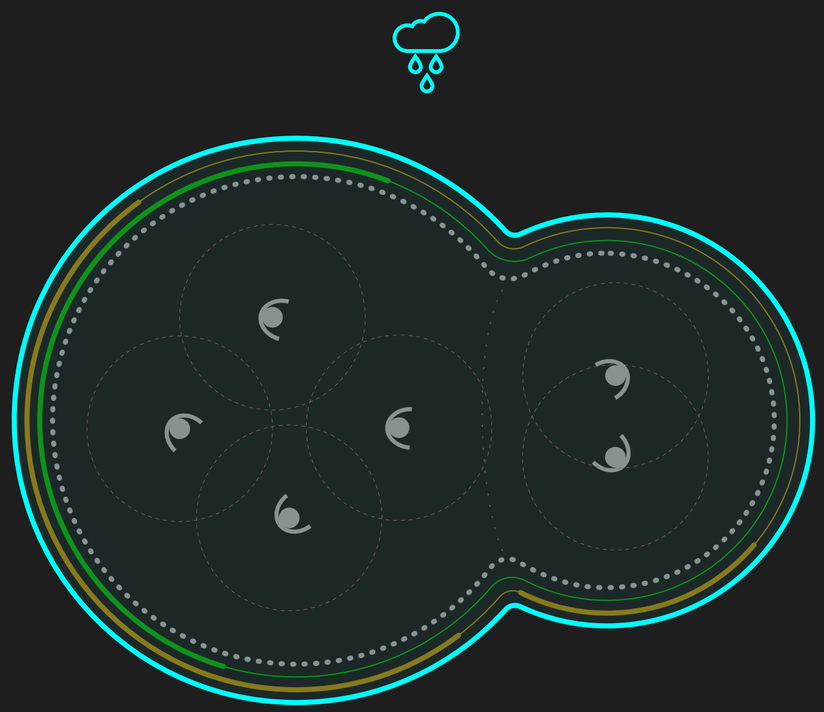
SHELTER LINE





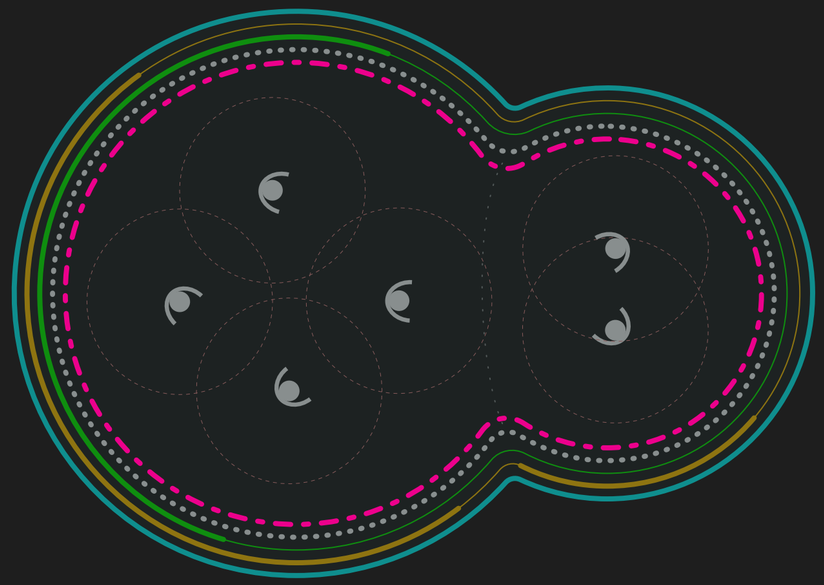
WIND


SUN



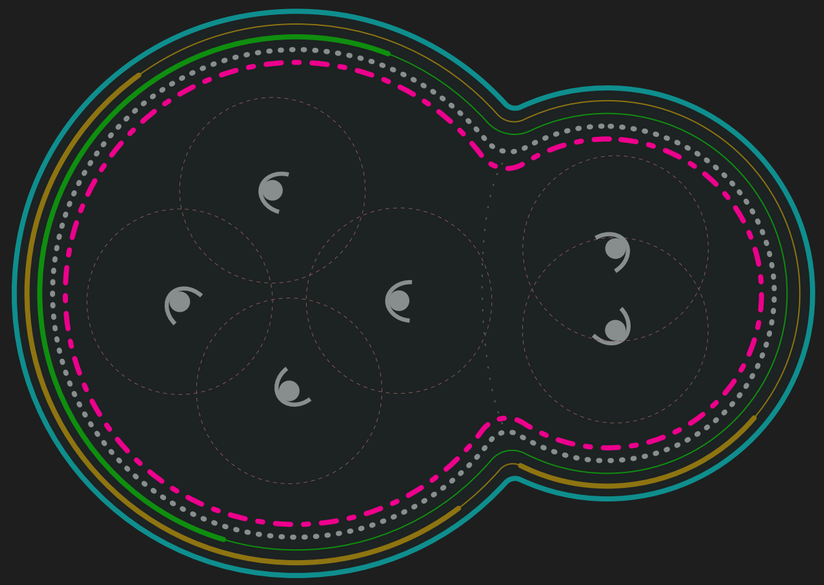


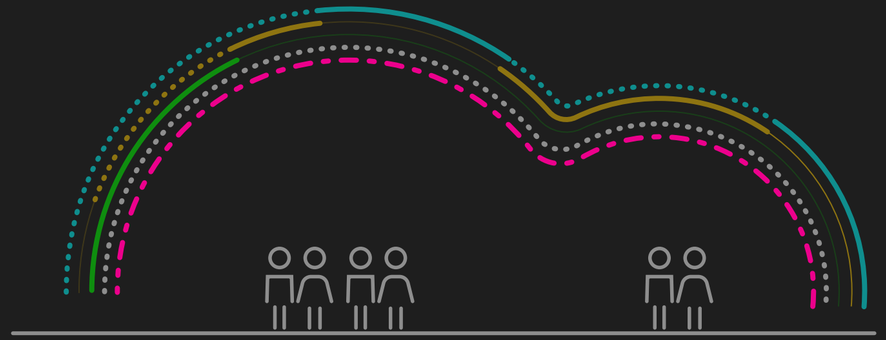
RAIN



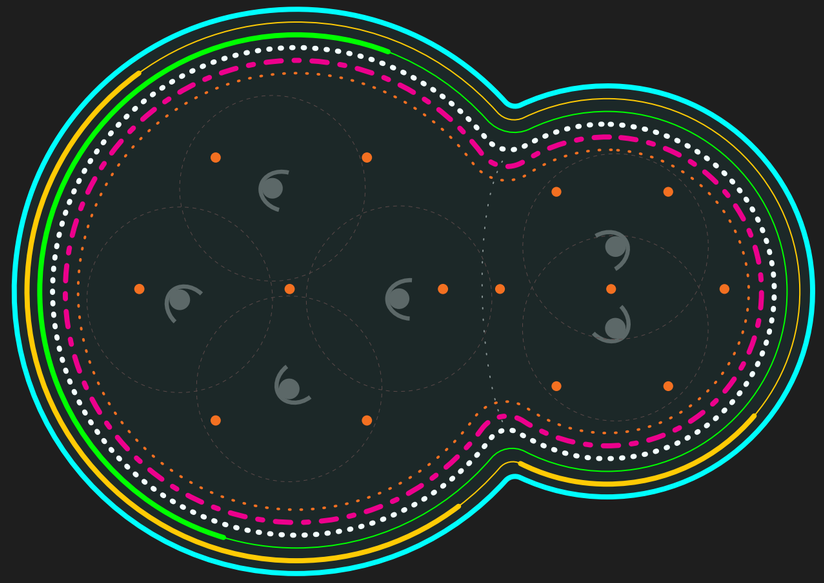


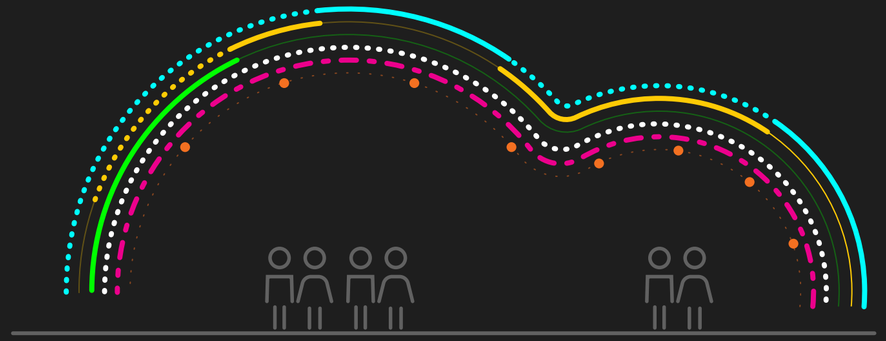
ACOUSTICS



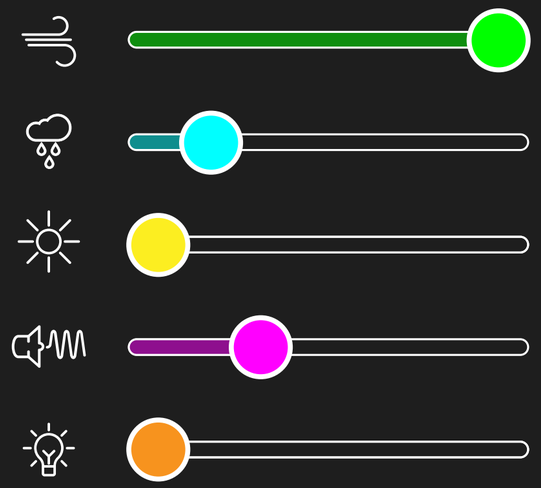


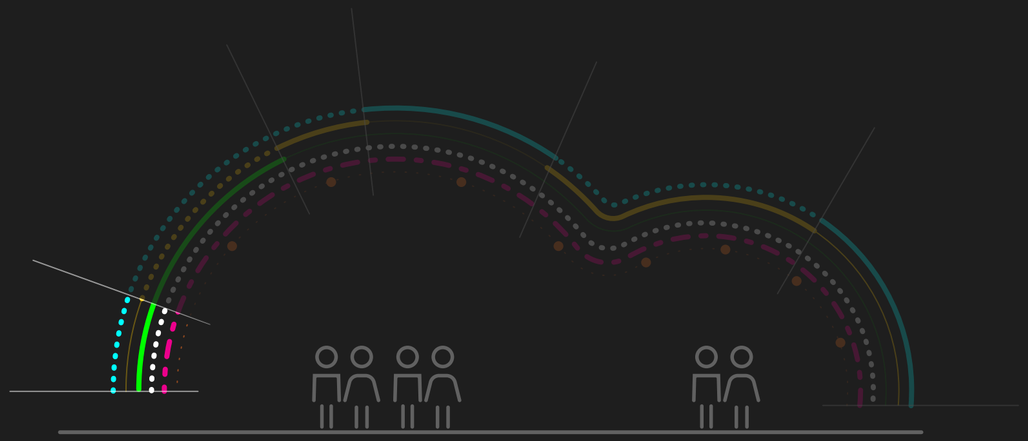
ARTIFITICAL LIGHTING

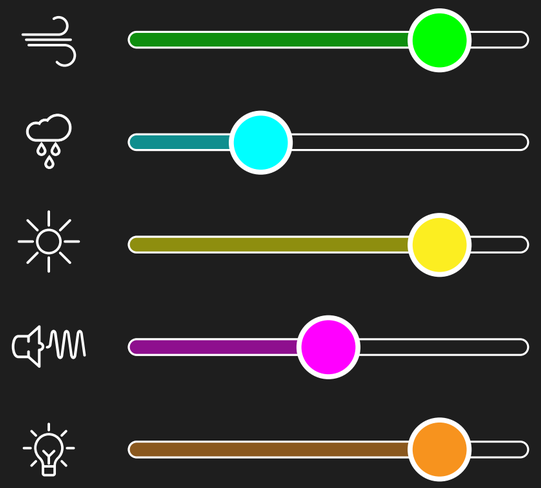




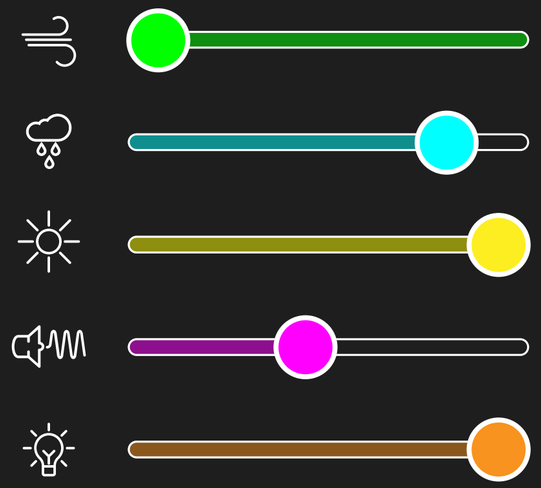
FINAL RESULT

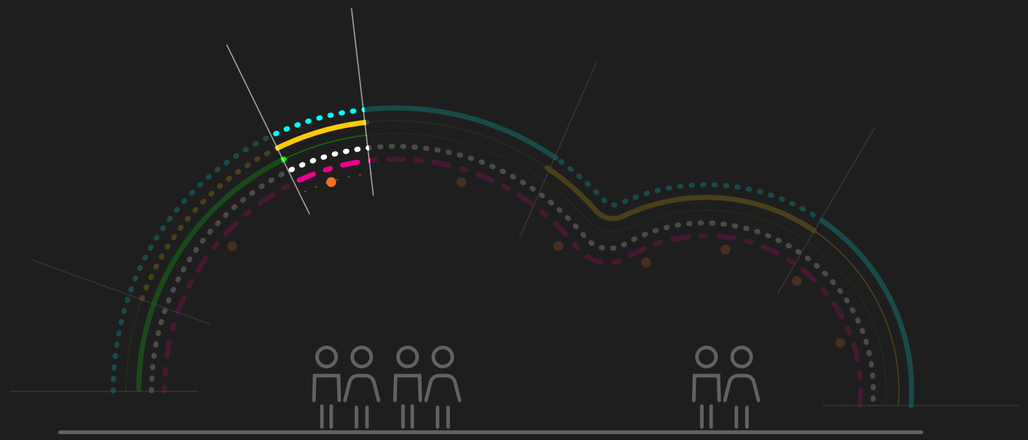


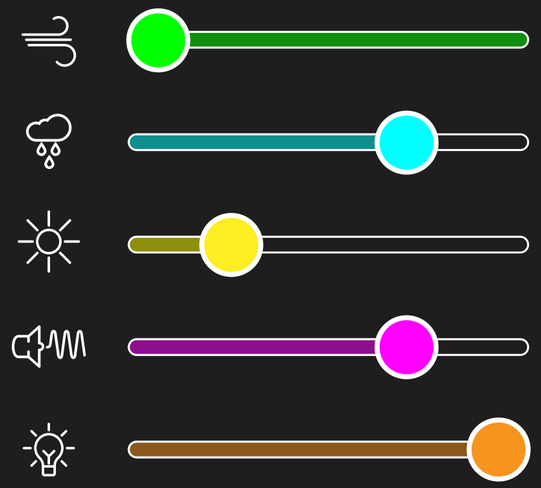


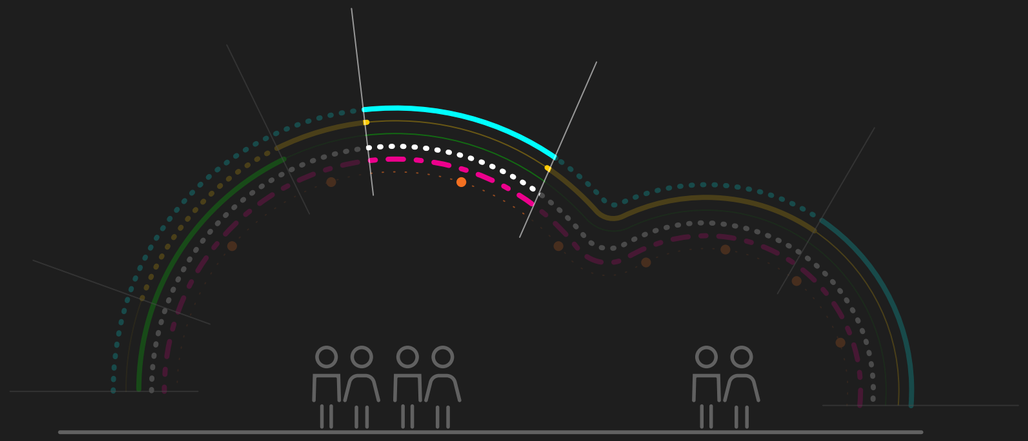


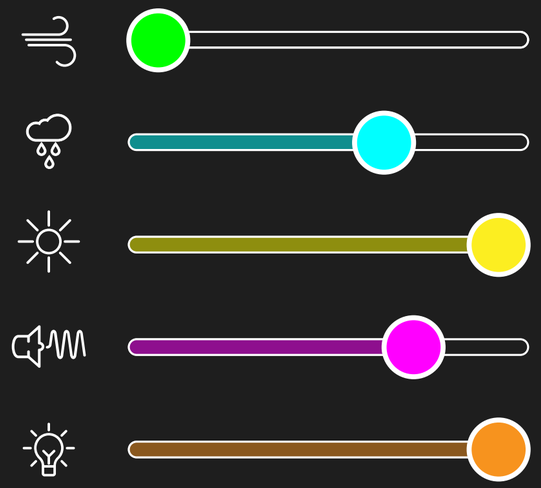


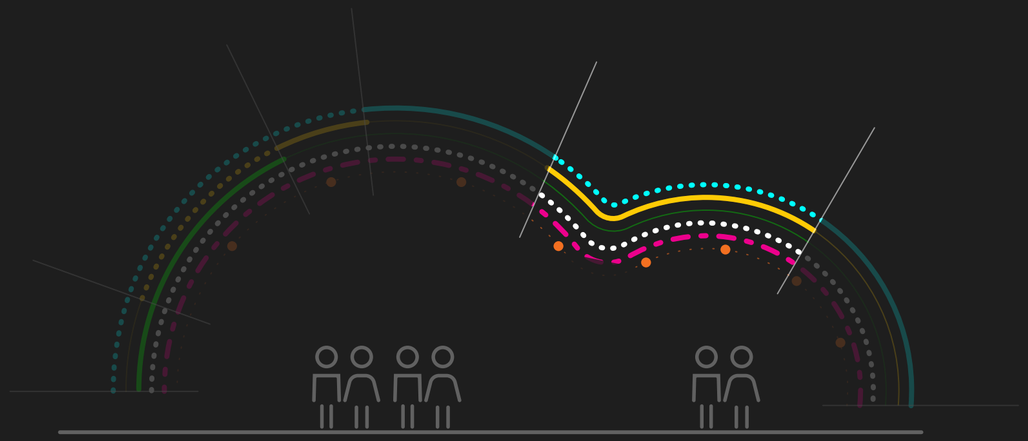


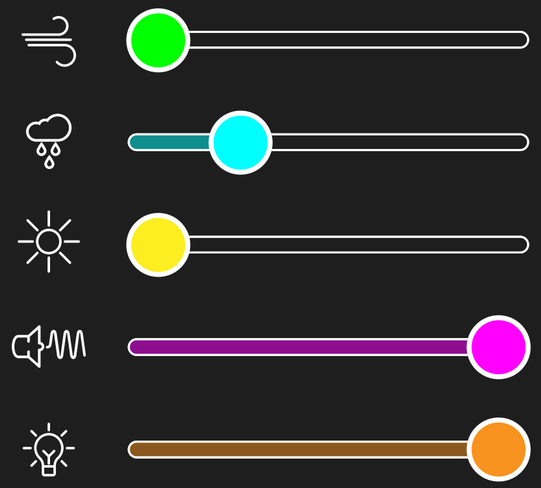


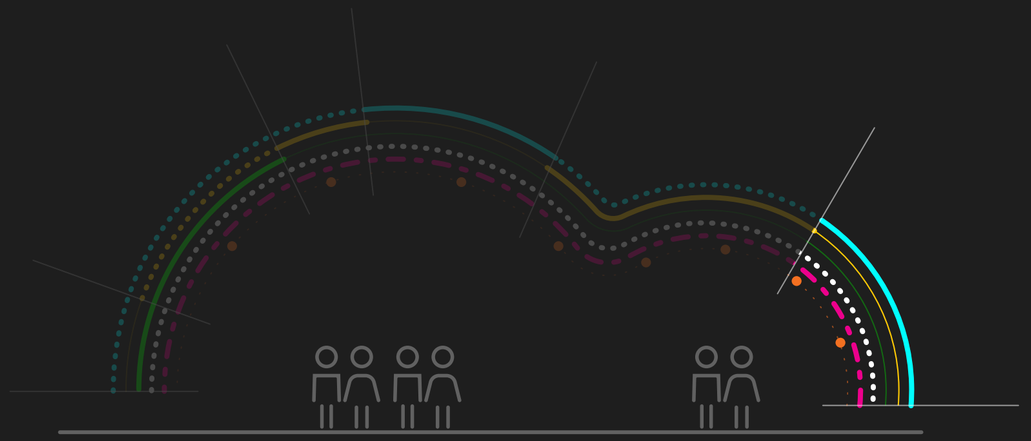






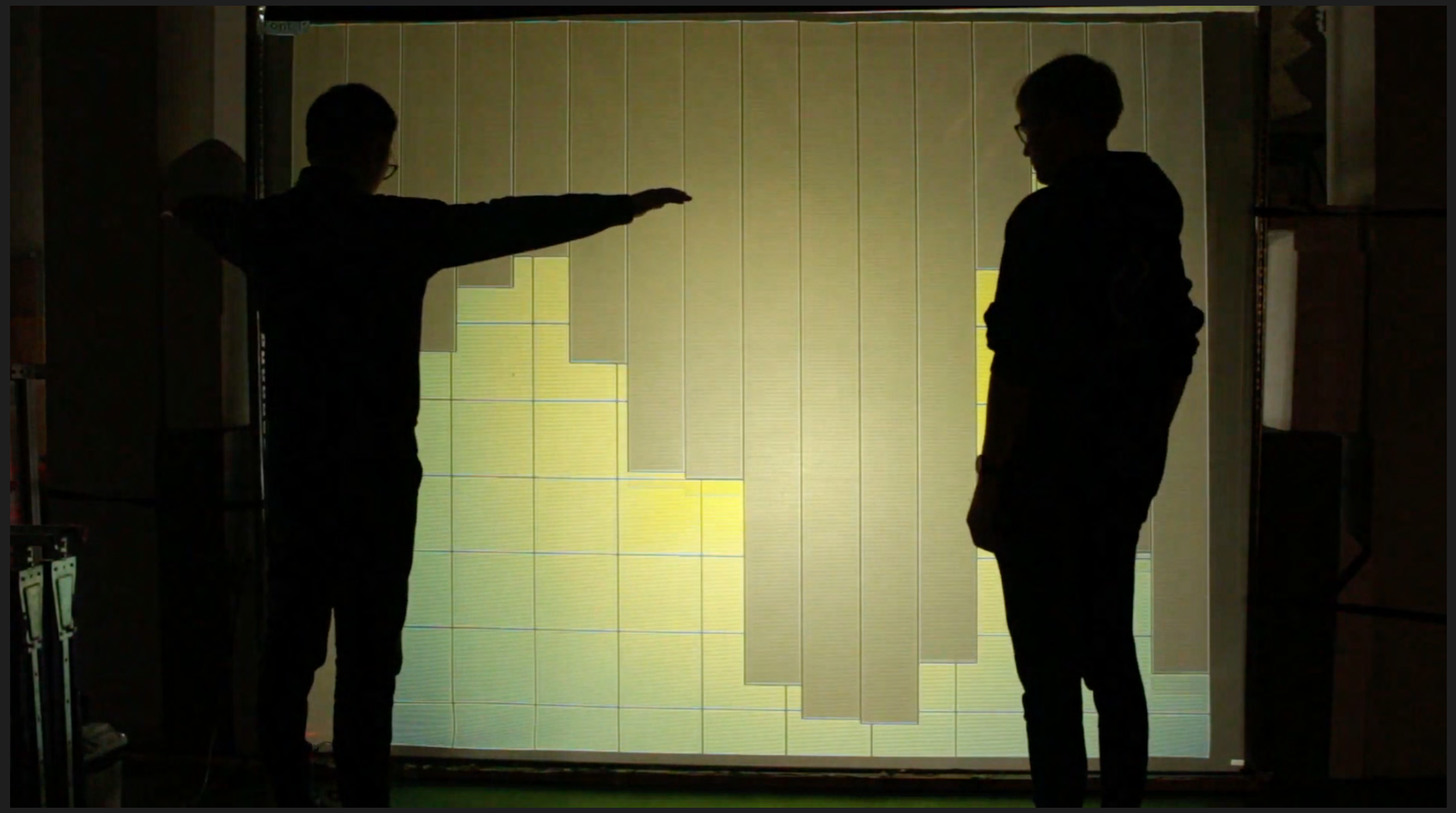






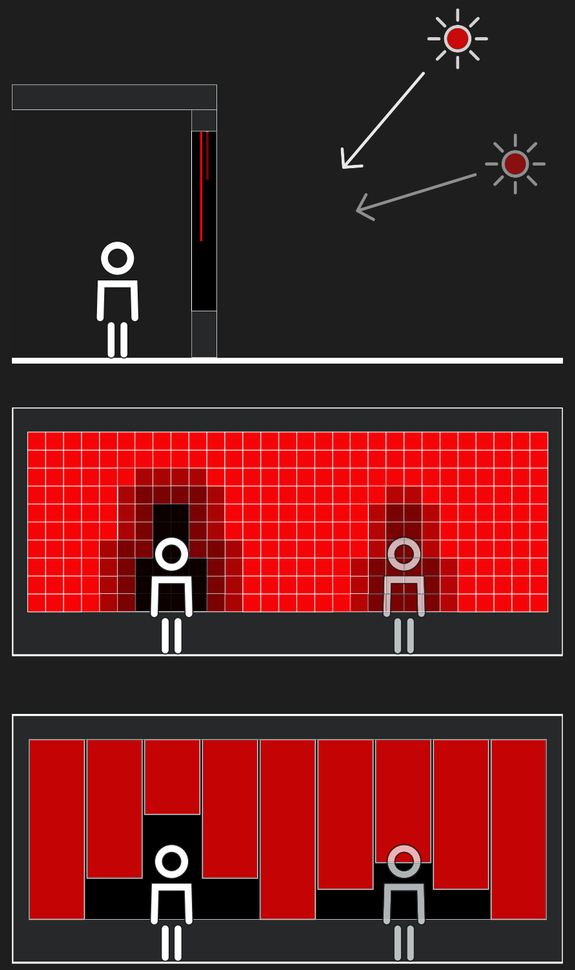

I'M SINGING IN THE RAIN!
OPEN 24 HOURS
A BRONX TALE
NASHVILLE
TOSHIBA
MARRIOTT MARQUIS
"ANDREW LLOYD WEBBER HAS BROADWAY ROCKING!"
Are you ready to wake up?
GREAT COMET
OUTERBORO
AMERICAN EAGLE OUTFITTERS
20 YEARS!
CHICAGOTHEMUSICAL.COM
McDonald's

PROTOTYPES

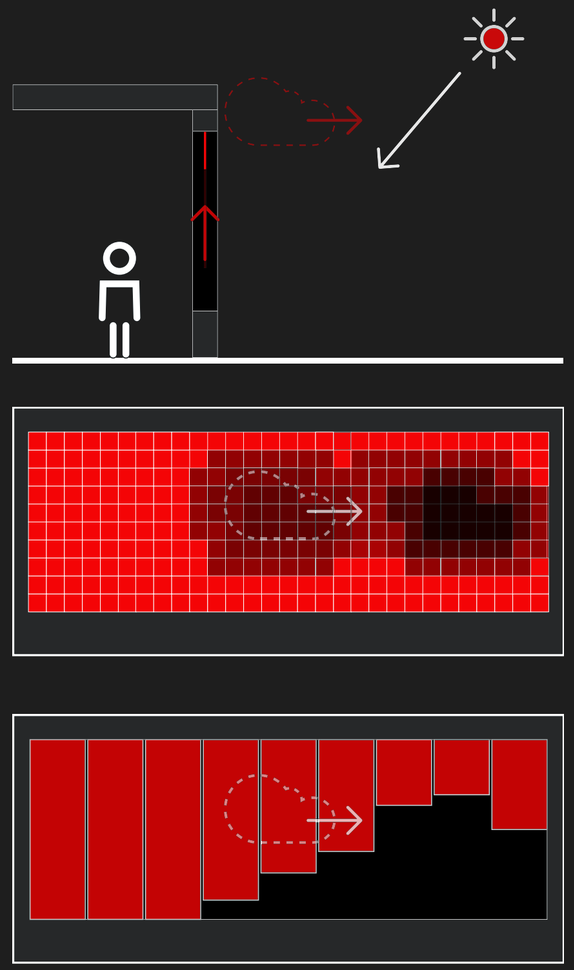


LOCAL+INTERACTIVE
SHADING SYSTEM AT BK

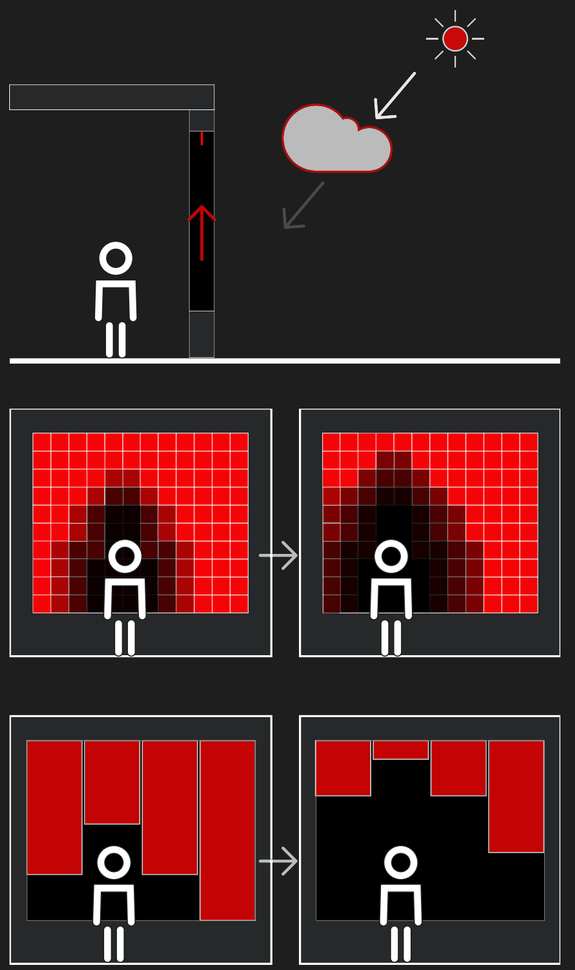
ENVIROMENTAL PARAMETERS



SEASONAL ADAPTATION

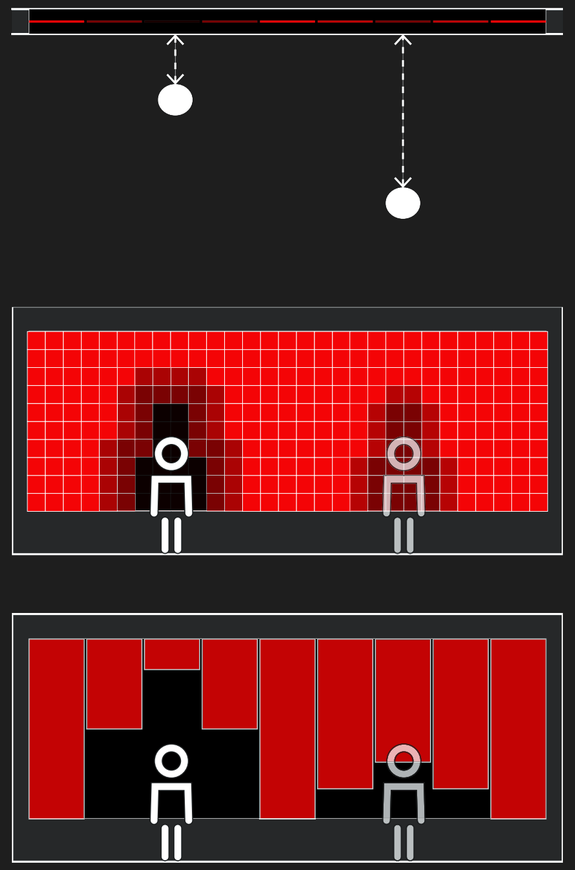


PREDICTIVE

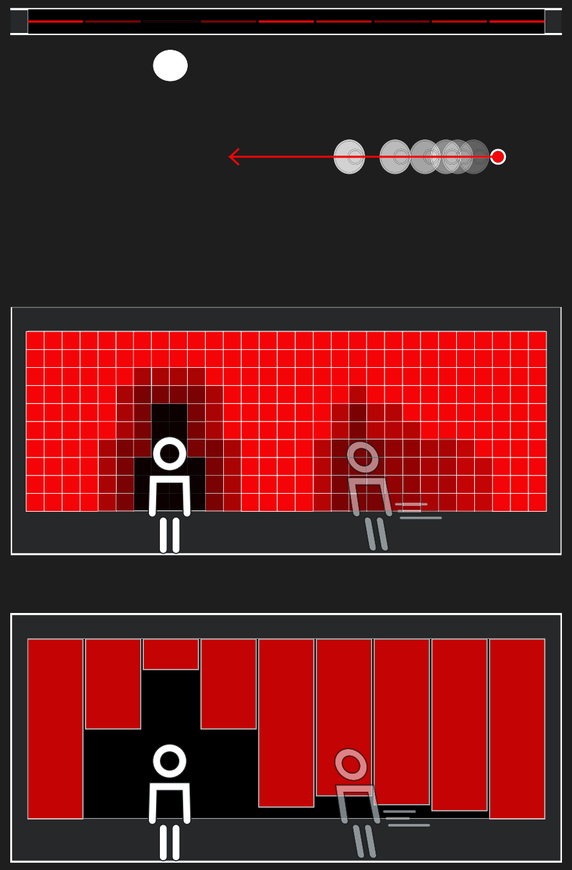


RESPONDS TO AMBIENT LIGHTING

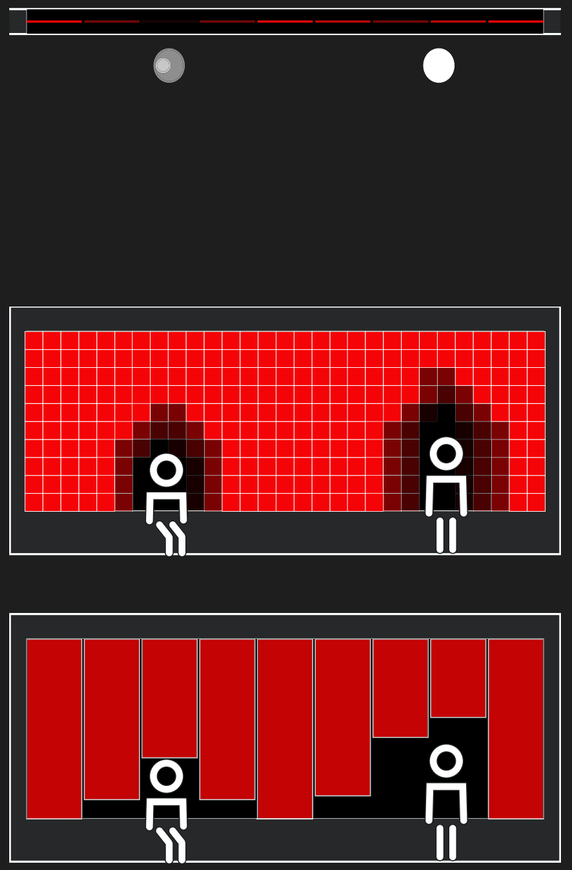
INTERACTIVE PARAMETERS



PROXIMITY TO THE WALL
INTENSIFIES THE REACTION



MOVEMENT ALONG THE WALL
VS STATIC POSITION



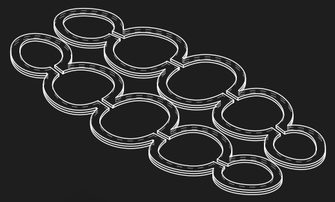
POSTURE: ENHANCED
COMFORT AND ACCURACY



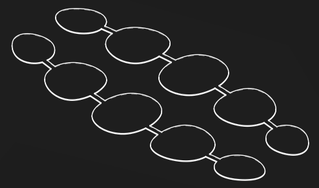


INFLATABLE SKIN
PROTOTYPE

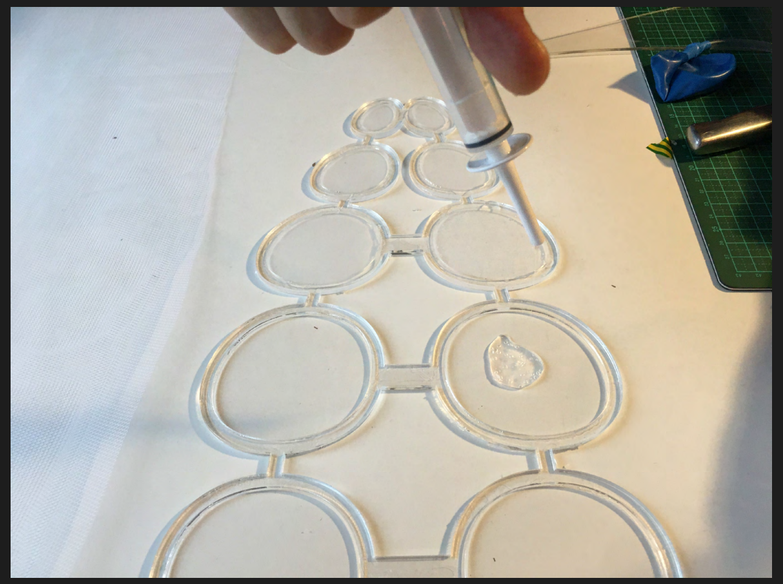
TOP SILICONE LAYER
2MM



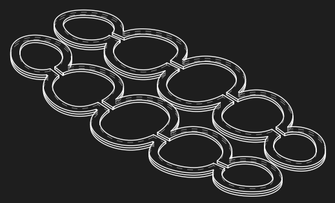
ACRYLIC MOLD



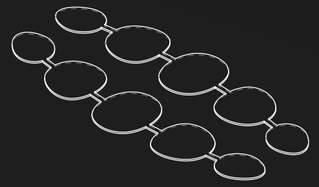
POUR SILICONE IN MOLD



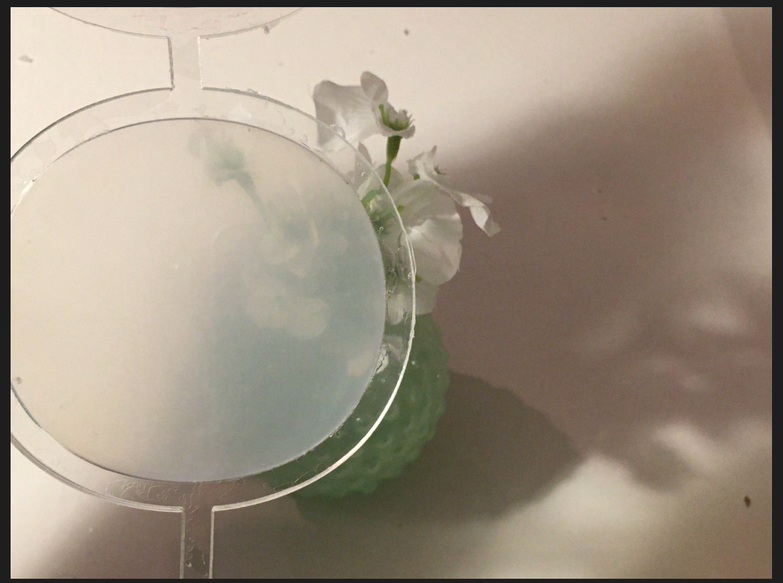
BOTTOM SILICONE LAYER
3MM

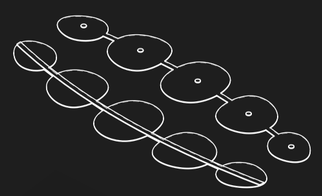


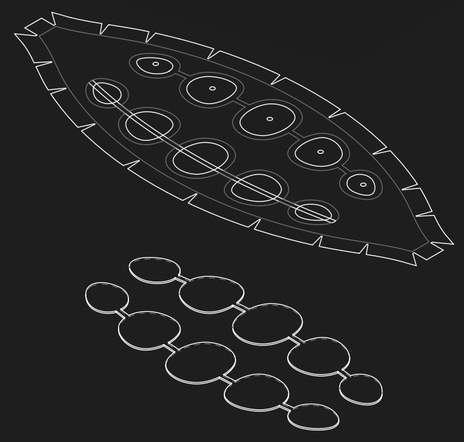
ACRYLIC MOLD



POUR SILICONE IN MOLD



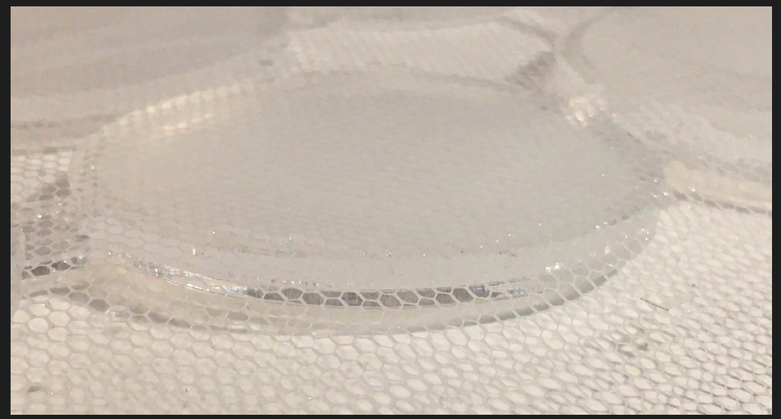


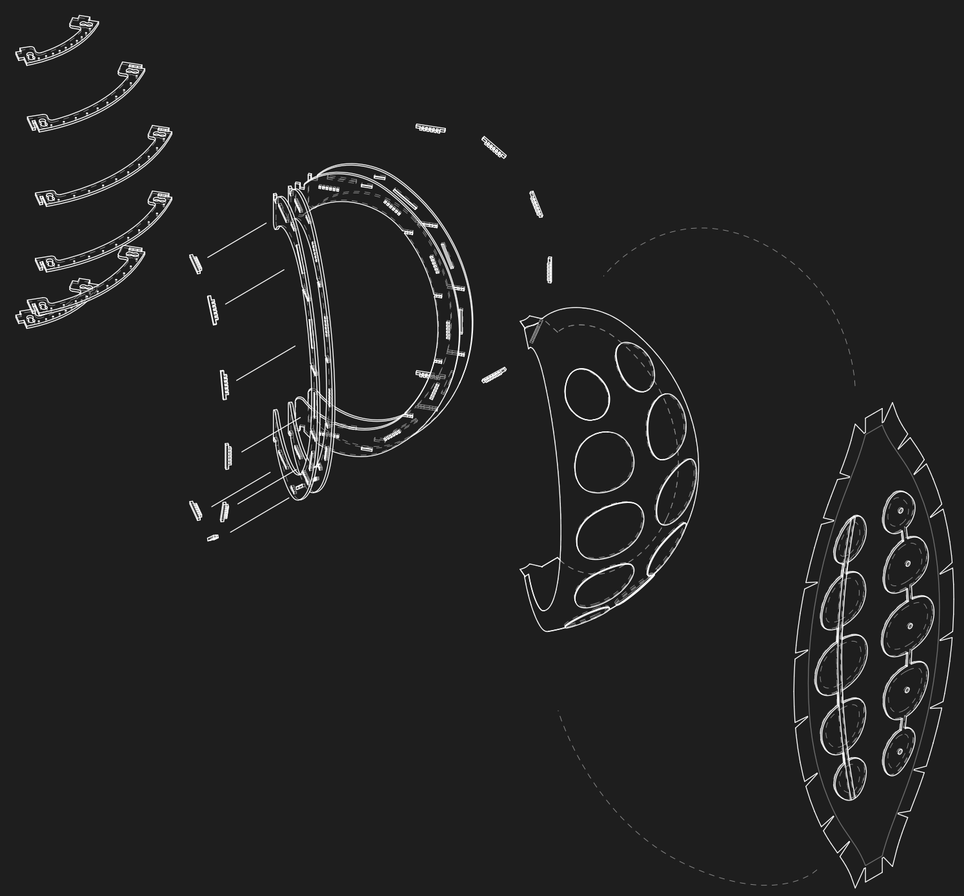


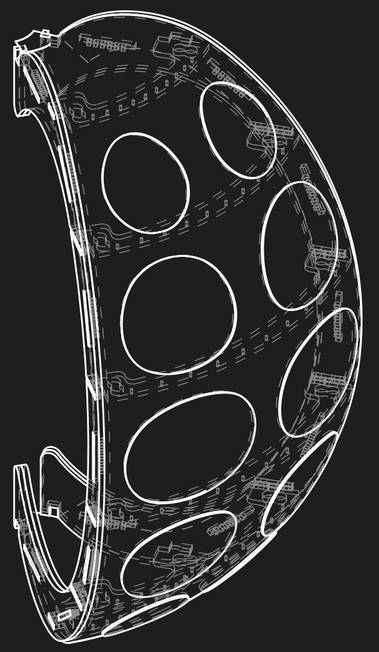
MESH FABRIC
0.1MM

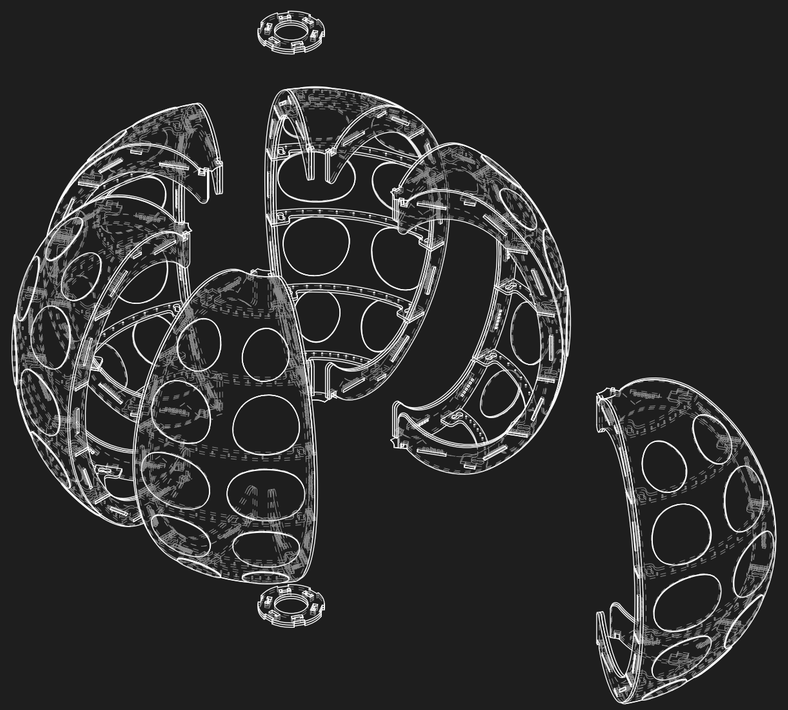
APPLY MORE SILICONE TO BOND
WITH TOP AND BOTTOM SILICONE LAYERS

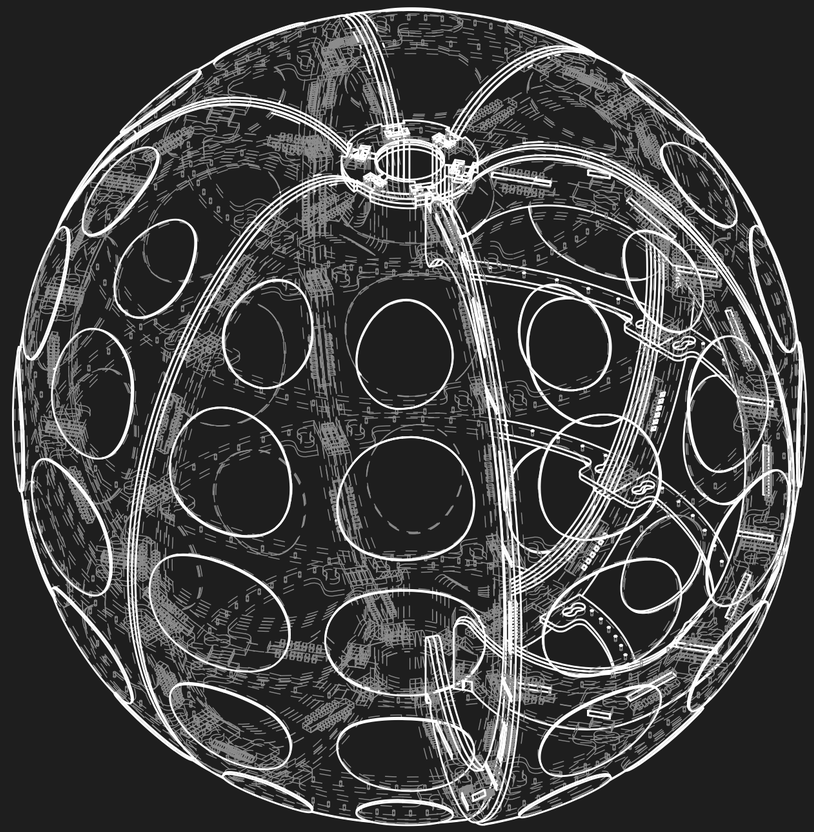




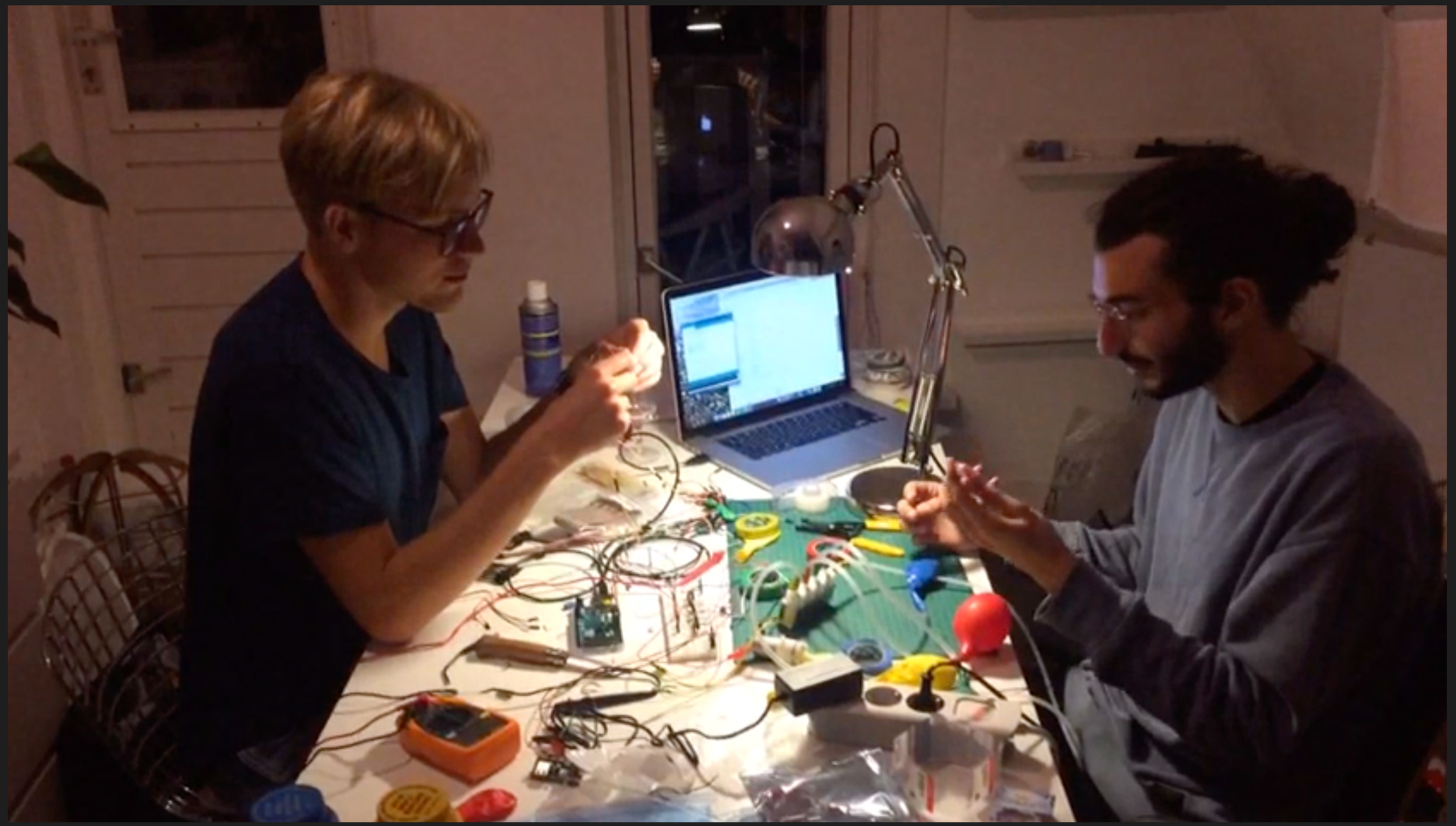






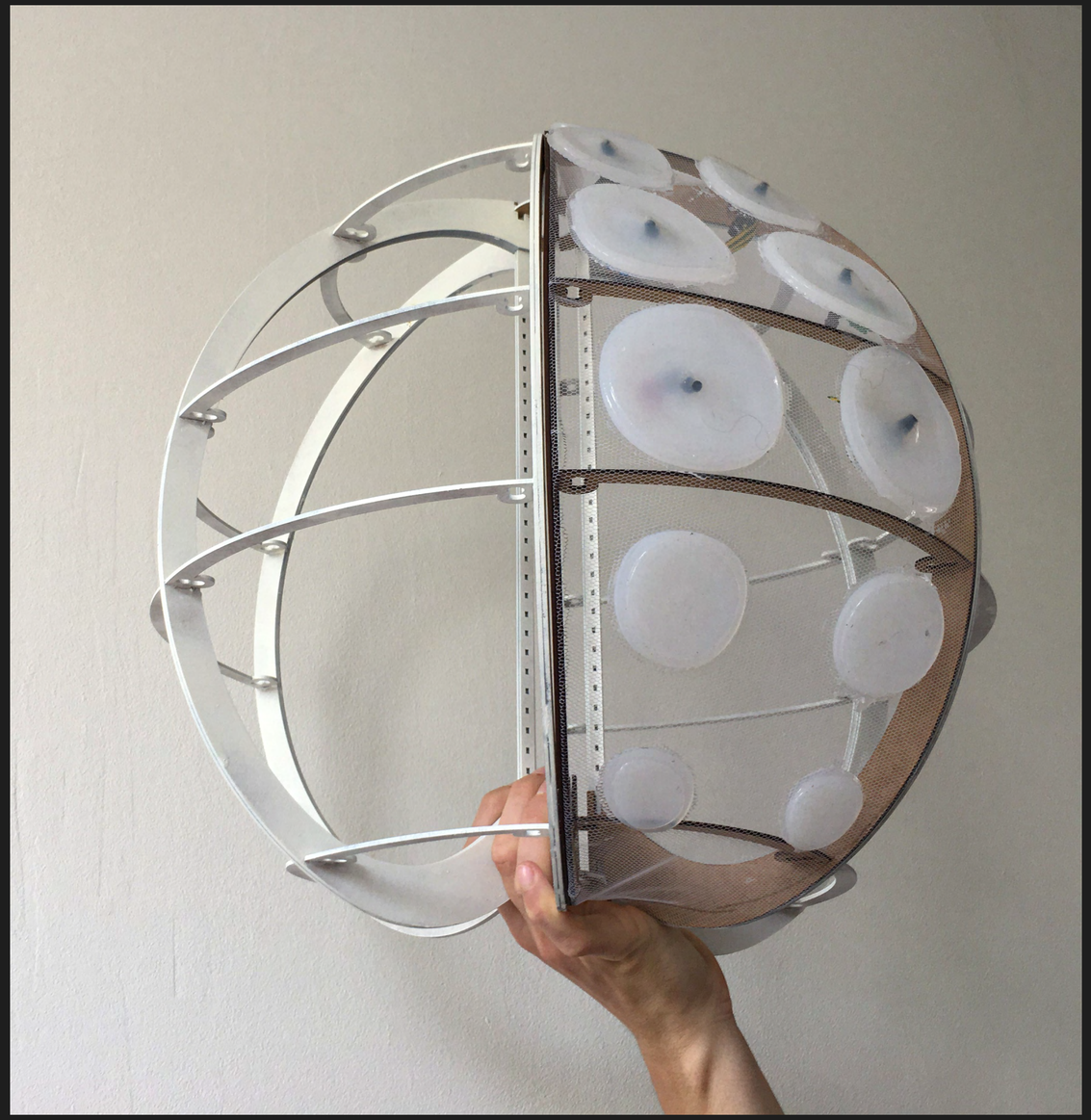












ARCHITECTURAL CONCEPT

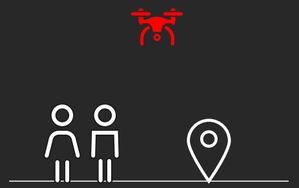




FORM TEMPORARY SHELTERS

RESPOND TO CLIMATE CONDITIONS

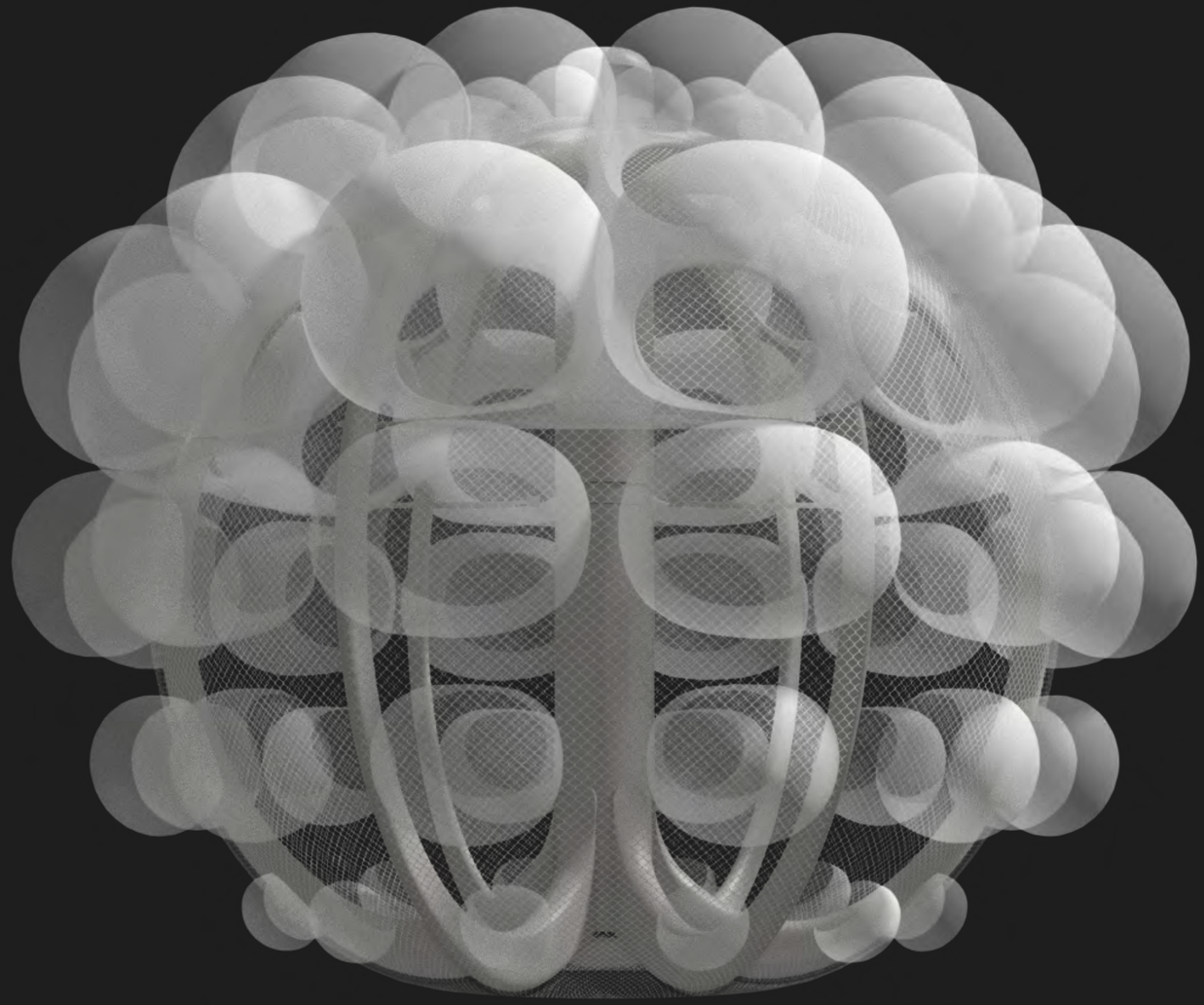
INTERACTION CONCEPT

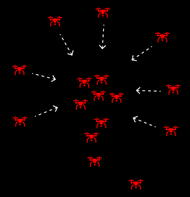
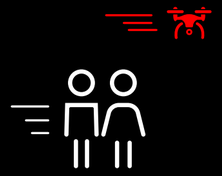
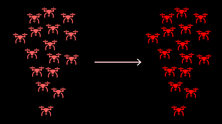
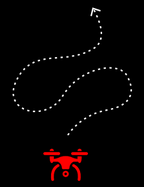
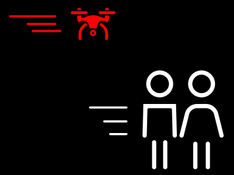
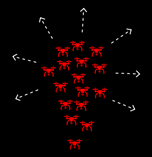
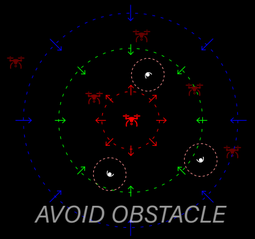
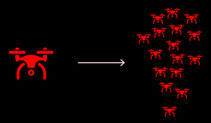
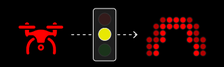




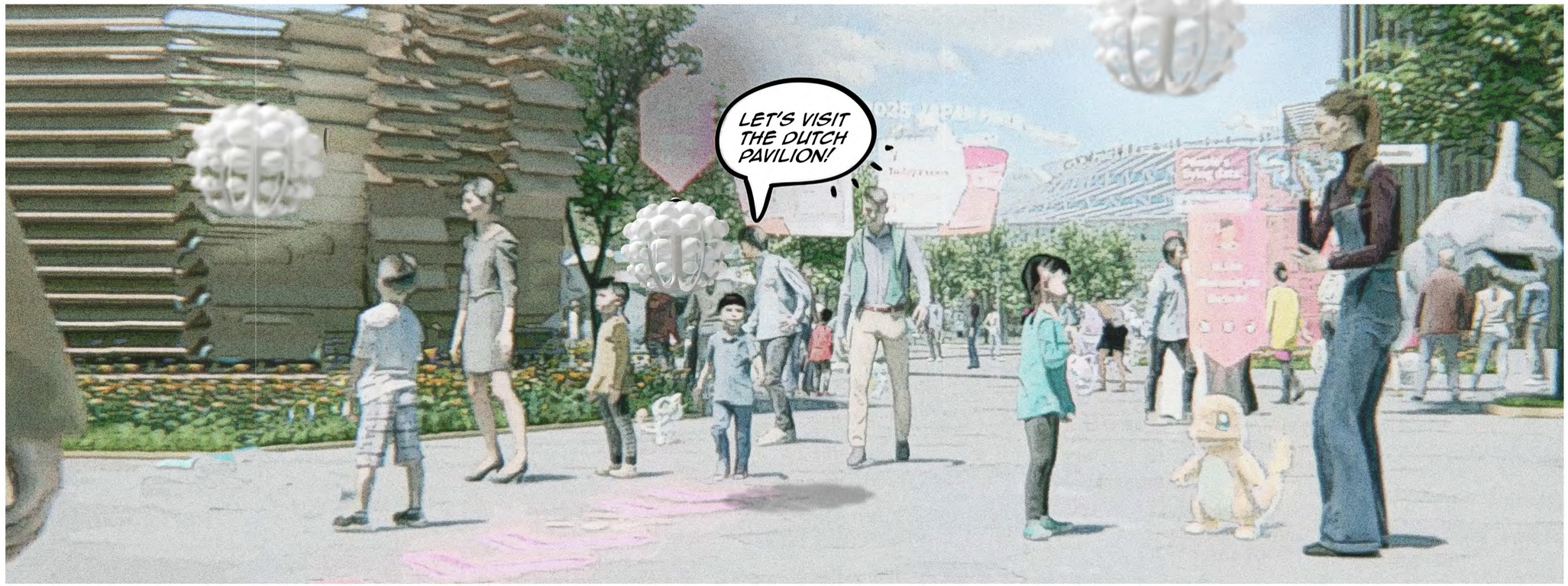
USER INTERACTION

PLAYFULNESS AND DELIGHT

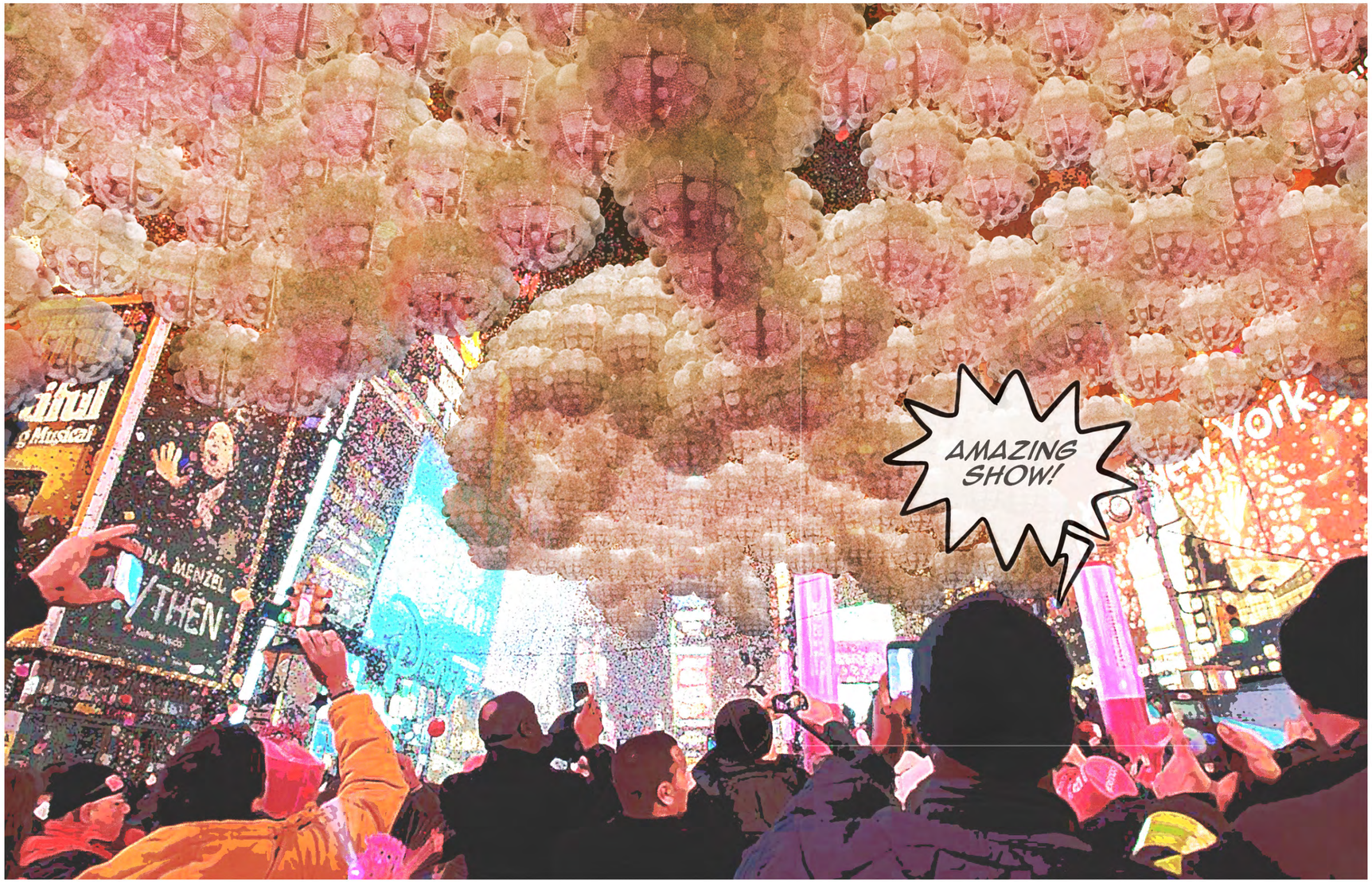


EMOTIONAL STATES	PERSONAL NEEDS	CREATURE INTERACTIONS	HUMAN INTERACTIONS	SWARM	ASSEMBLY
 FEELINGS	 CHARGE	 FORM A SWARM	 GUIDE	 MERGE WITH OTHER SWARM	 FORM ASSEMBLY
 FOCUS	 WANDER	 HELP	 FOLLOW	 RELEASE SWARM	 MORPH ASSEMBLY
 INTEREST	 AVOID OBSTACLE	 CONNECT	 REPOSITION	 MARK LOCATION	 MODULATE ENVIRONMENT
	 PLAY	 SHARE INFORMATION	 LIGHT	 LOOK FOR CROWD	 RESPOND TO HUMAN INPUT
	 JOIN A SWARM				 HOVER / STANDBY

BEHAVIOURS


LET'S VISIT THE DUTCH PAVILION!


I'M SINGING IN THE RAIN!
OPEN 24 HOURS
A BRONX TALE
NASHVILLE
TOSHIBA
MARRIOTT MARQUIS
"ANDREW LLOYD WEBBER HAS BROADWAY ROCKING!"
SCHOOL of
Are you ready to wake up?
GREAT COMET
AMERICAN EAGLE OUTFITTERS
20 YEARS
CHICAGOTHEMUSICAL.COM
OUTFRONT


AMAZING SHOW!
York
NA MENZEL
THEN



