

RhinoScripting Workshop :: Lima 2008 :: www.espaciosdigitales.org/lima/1

GRUPO 2 |

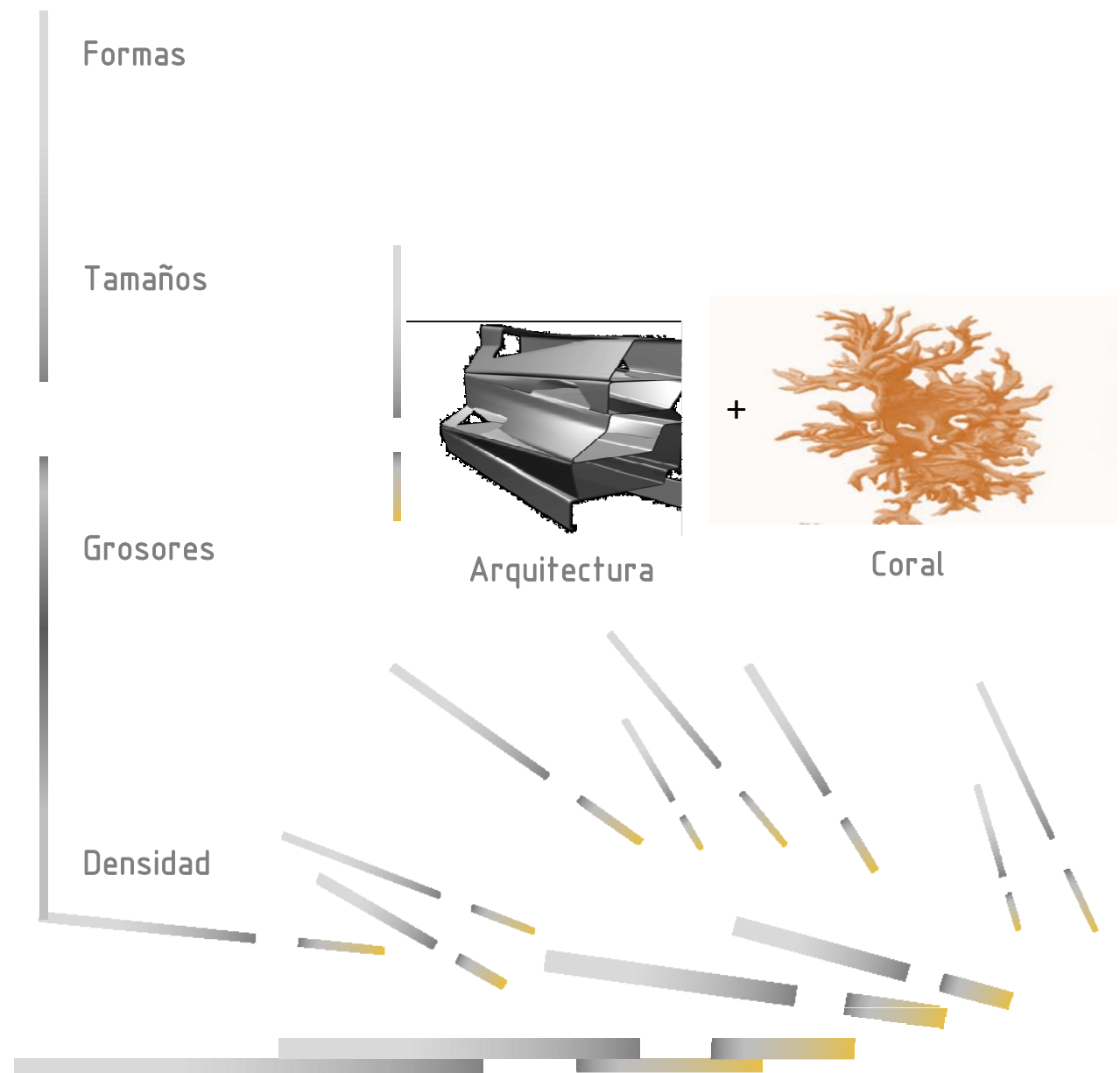
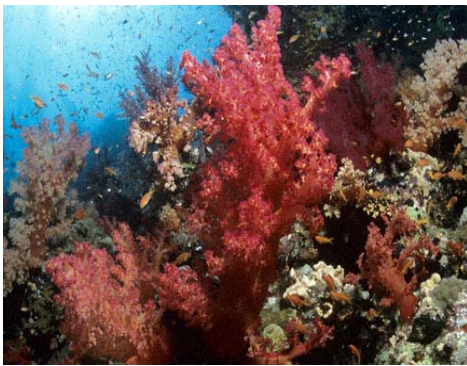
Andrea Jimenez

Claudia Soria

Julio Miguel Saquiray



CORAL BRANCHES ARCHITECTURE



CORAL-ORIGEN DE LA IDEA

_El_Script_Parte1

_ El usuario comienza ubicando y dibujando el troco, que es el punto de partida.

_ El script se trabaja, de una forma simple al comienzo. Calculando a través de formas trigonométricas, como dibujar cada una de las 4 ramas que salen del troco en un cuadrante diferente.

```
++ 'Get the EndPoint of the trunk
endPt = Rhino.curveEndPoint(treeTrunk)
Rhino.AddPoint endPt
Rhino.Print "X coord: " & endPt(0)
```

_ La posición de cada una de estas ramas está comprendida en una función "random" que nos permite obtener árboles únicos.

```
++ Creating the points parameters
point1(0)=treeEnd(0)+ abs((distTrunk/3)*cos(AngRand))
point1(1)=treeEnd(1)+ abs((distTrunk/3)*sin(AngRand))
point1(2) = Rnd() + locZ + distTrunk/5

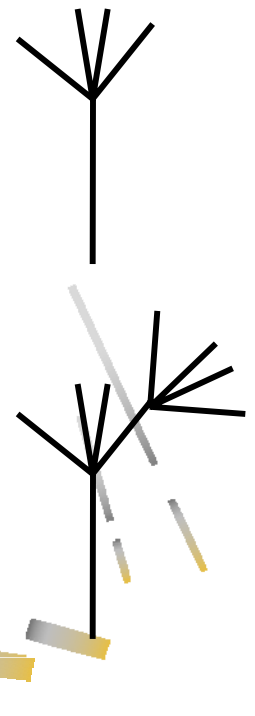
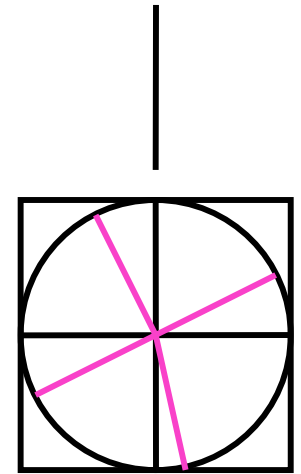
Rhino.Addpoint point1
```

_ Por otro lado se crea una función que nos permite determinar progresivamente la medida de la siguiente rama por crecer. Esta función está en relación con la altura del tronco o rama anterior. La relación nos permite que durante el crecimiento la rama nueva vaya disminuyendo en medida.

_ Nos interesa colocar un límite de crecimiento. Éste dependerá de la medida mínima a la que puede llegar una rama. Ej. La menor debe de tener max una long. De 0.5cm.

_ Una vez armadas estas funciones, debemos de llamarla "call" con el fin de que se repita.

```
++ Funcion Recursiva
If (Rhino.Distance(Rhino.CurveEndPoint(branch1) , Rhino.CurveStartPoint(branch1 )) > (globalSize/3)) Then
    CALL div1(branch1)
    Call div1(branch2)
    CALL div1(branch3)
    Call div1(branch4)
```



¿CÓMO LO LOGRAMOS?

_El_Script_Parte2

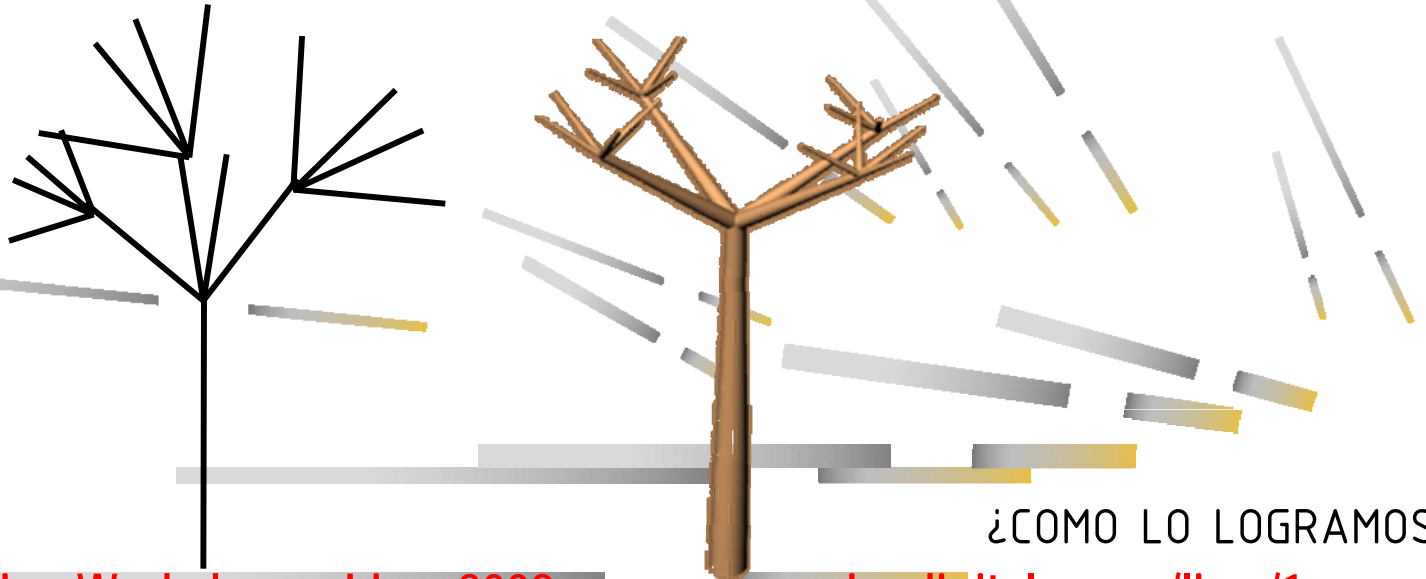
_Una vez que obtenemos la estructura del coral de manera lineal, pasamos a darle volumen.

_El volumen del coral se obtiene a través de la función "pipe" o tubo.

```
botSizeb4 = Rhino.CurveLength(branch4)/20  
topSizeb4 = Rhino.CurveLength(branch4)/40  
  
'piping branch4  
Rhino.SelectObject branch4  
Rhino.Command "Pipe " & CStr(botSizeb4) & " " & CStr(topSizeb4) & " " & " "  
Rhino.UnselectAllObjects()
```

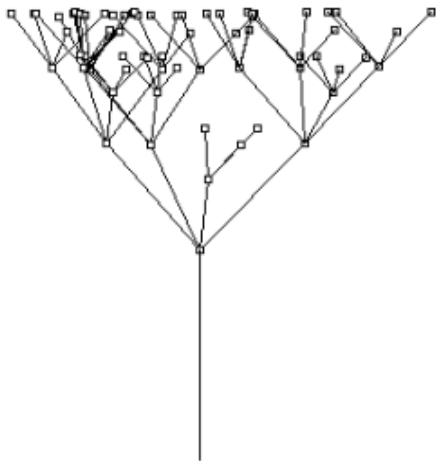
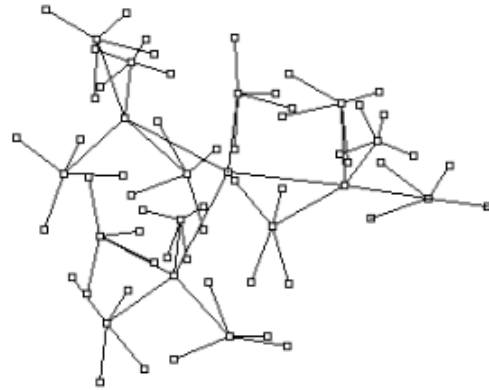
_Esta función estará conformada por dos partes, el diámetro de origen y el de termino. Estos diámetros estarán en proporción al tamaño del árbol, para que se pueda ir reduciendo en relación al tamaño total.

_La función nos generará la siguiente rama teniendo en cuenta el diámetro de la rama desde la cual se originó.

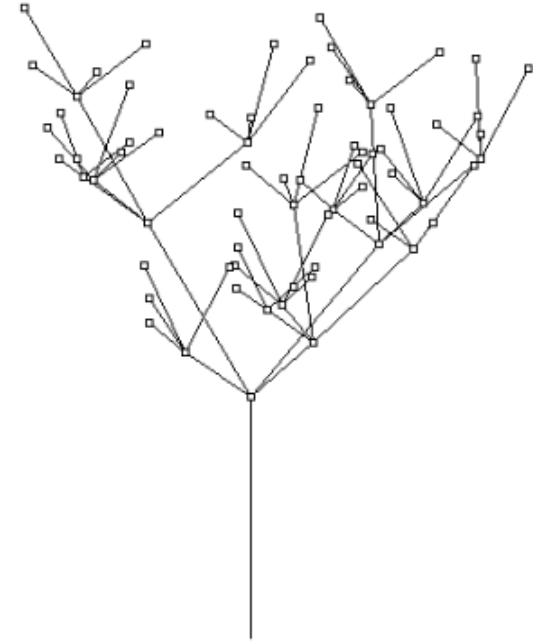


_Video_Crecimiento_Lineal

perCam 2



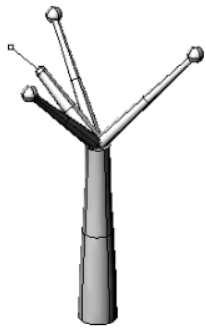
Perspective



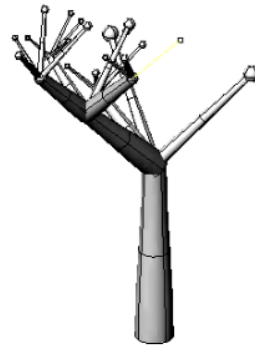
ANIMANDO

_Video_Crecimiento_3D_"pipe"__

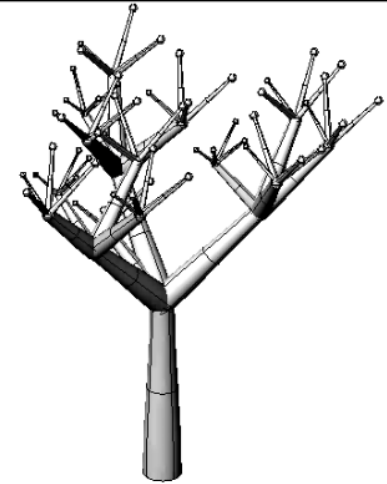
Perspective



Perspective

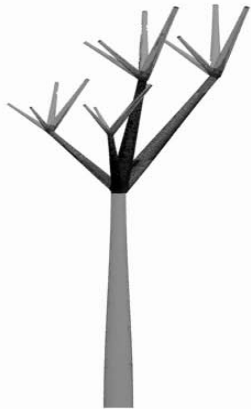


Perspective



ANIMANDO

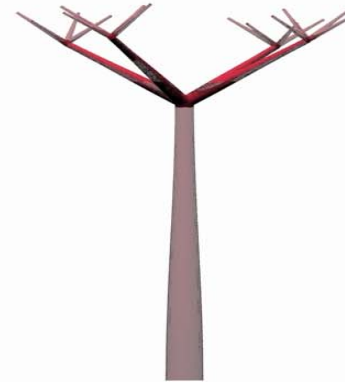
Posibles_Opciones_Generadas_por_Script



Se aumento el punto z
A 2 ramas (3)



Se redujo el factor de
crecimiento a 2



Se vario el punto Z a 1.4
(+ abierto)



Se vario el punto Z a 5
(+ estrecho)



Se vario el punto Z a 5
(mas ramificación y altura)



Se vario el punto Z a 5
Para dos ramas y a 2 para
las otras 2



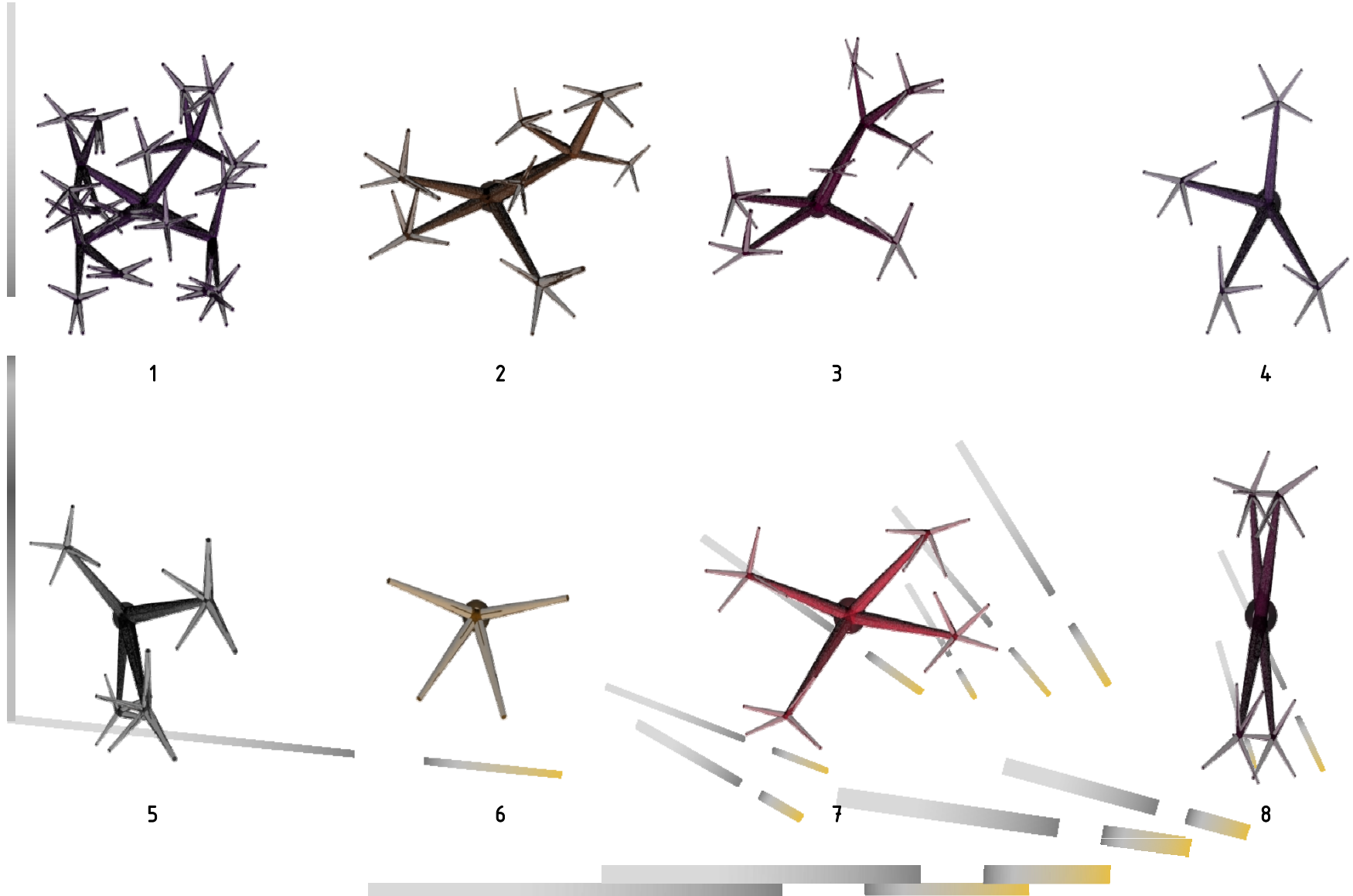
Anterior + aumento del
factor global a 4 (mas
ramas)



Reduccion de crecimiento
global y Z

¿QUE OPCIONES TENEMOS?

Posibles_Opciones_Generadas_por_Script / Top



¿QUE OPCIONES TENEMOS?

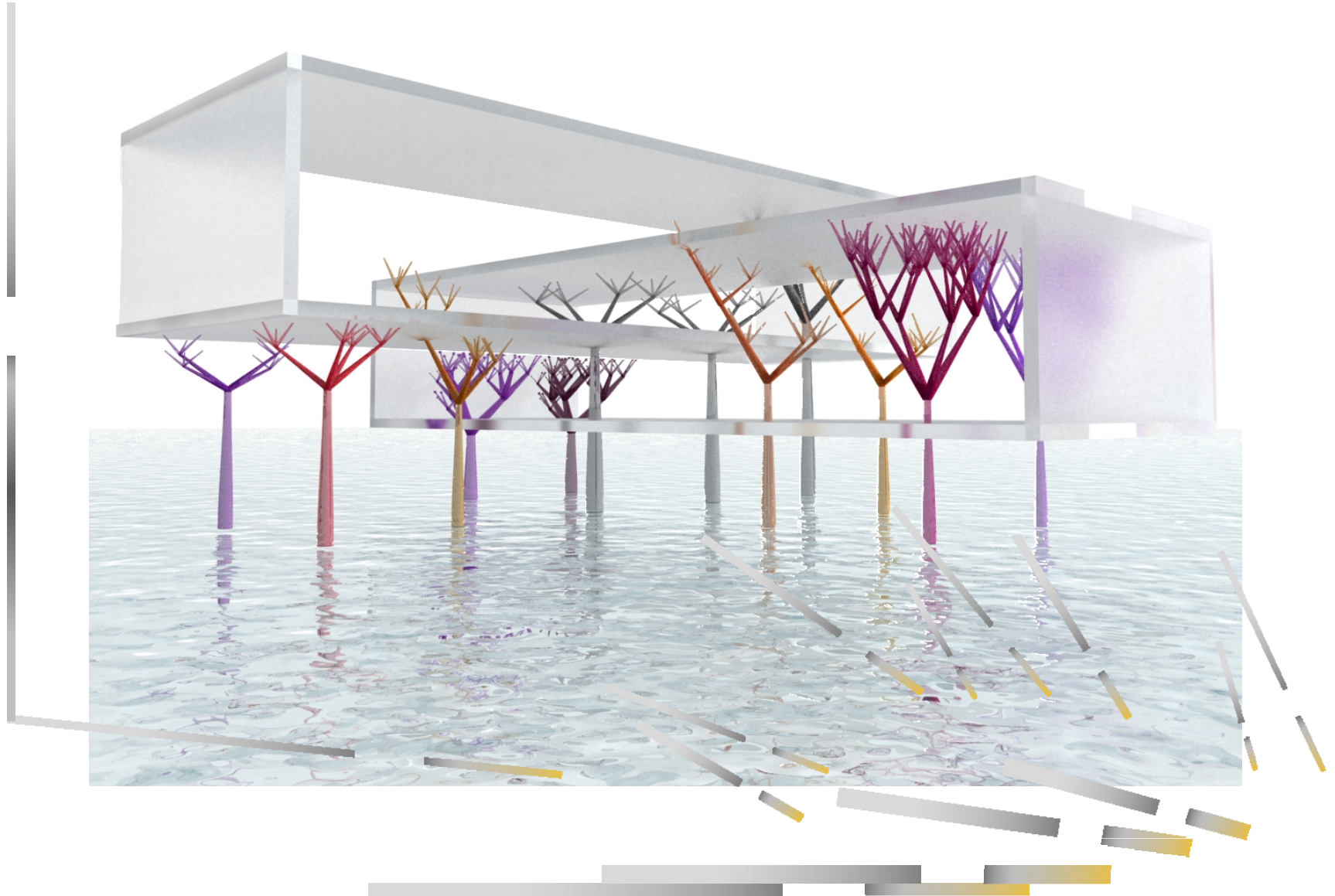
Posibles_Opciones_Generadas_por_Script / 3D



¿QUE OPCIONES TENEMOS?

RhinoScripting Workshop :: Lima 2008 :: www.espaciosdigitales.org/lima/1

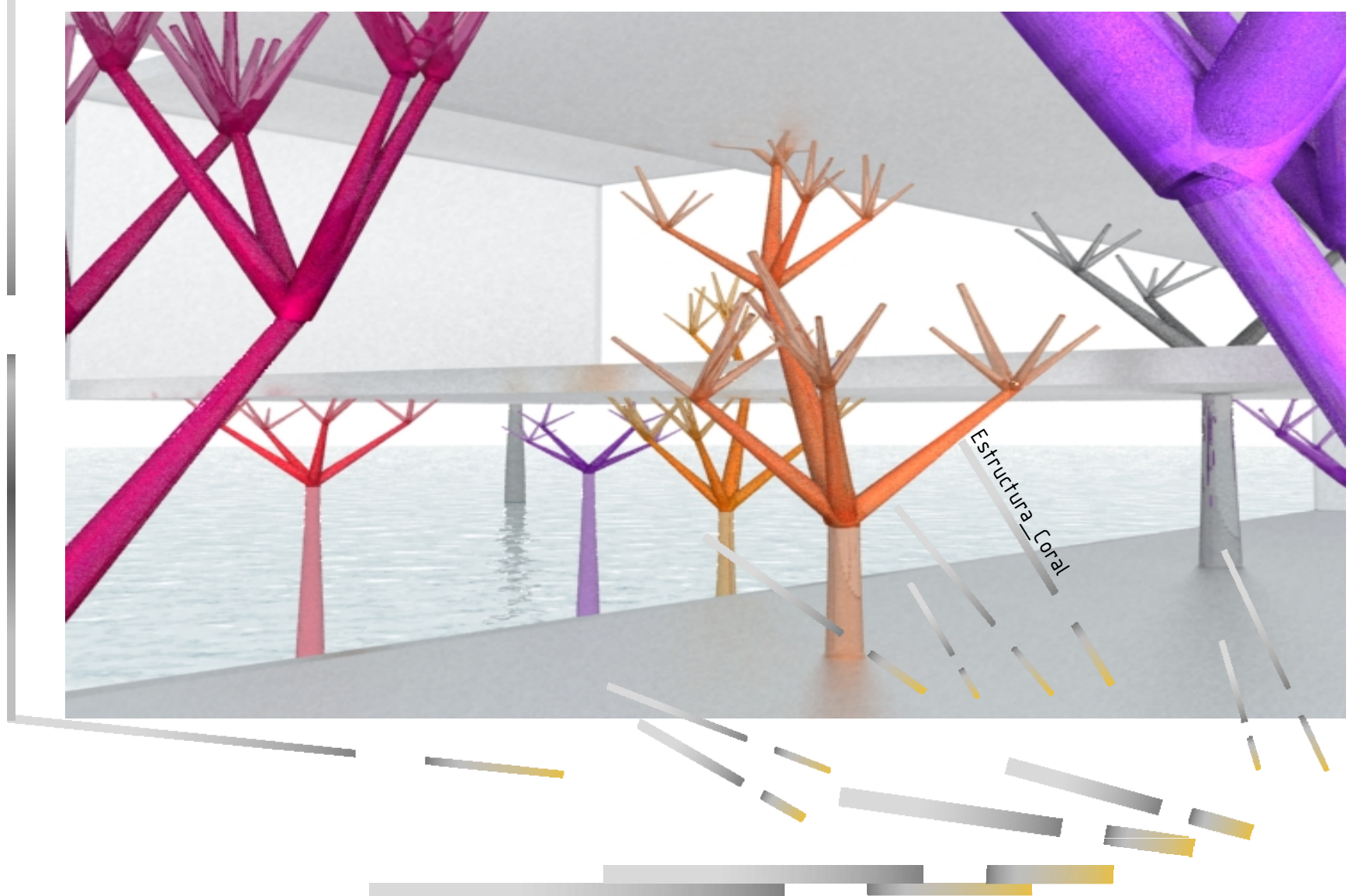
Resultado



CONFORMANDO LA IDEA

RhinoScripting Workshop :: Lima 2008 :: www.espaciosdigitales.org/lima/1

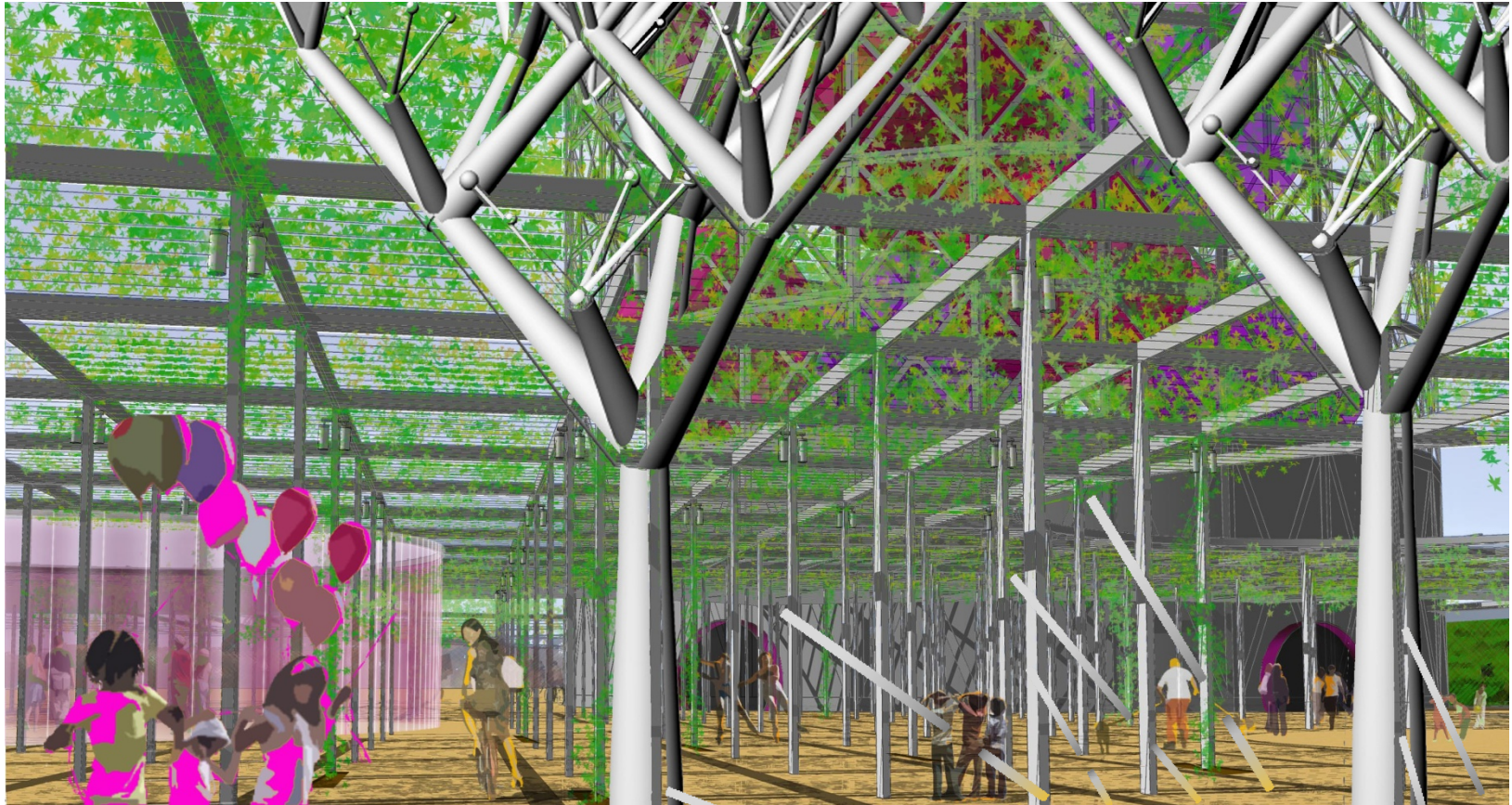
Objetivo = corales como estructura.



CORALES COMO ESTRUCTURA

RhinoScripting Workshop :: Lima 2008 :: www.espaciosdigitales.org/lima/1

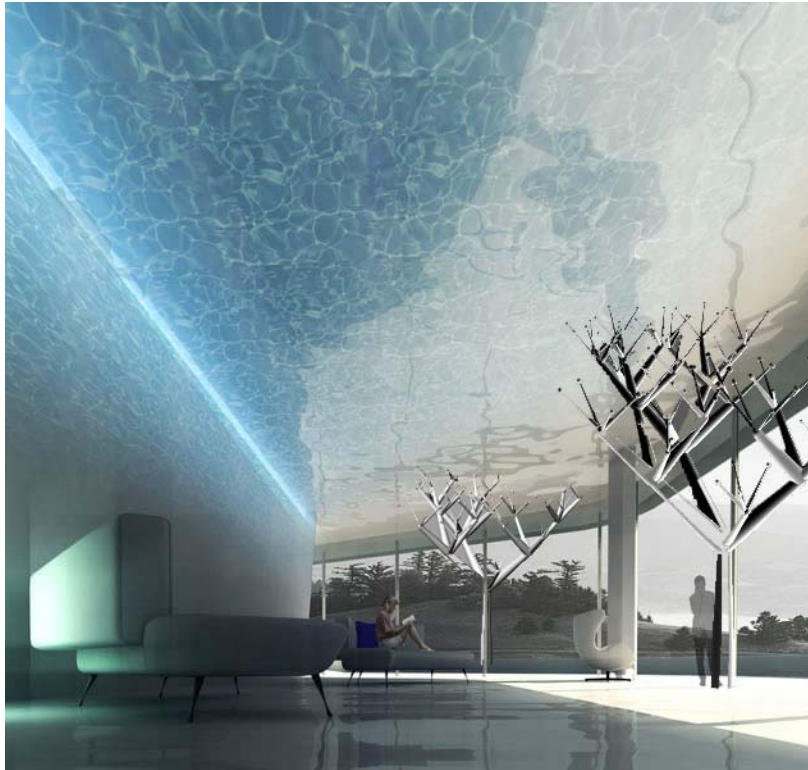
Objetivo = corales como estructura.



CORALES COMO ESTRUCTURA

RhinoScripting Workshop :: Lima 2008 :: www.espaciosdigitales.org/lima/1

Jugando con las Diversas Opciones



COMPLEJO DEPENDE DE LA VARIEDAD

RhinoScripting Workshop :: Lima 2008 :: www.espaciosdigitales.org/lima/1

_Más_Arquitectura



OTRAS CREACIONES ARQUITECTÓNICAS

RhinoScripting Workshop :: Lima 2008 :: www.espaciosdigitales.org/lima/1

Más Arquitectura



OTRAS CREACIONES ARQUITECTÓNICAS

RhinoScripting Workshop :: Lima 2008 :: www.espaciosdigitales.org/lima/1

Más Arquitectura



OTRAS CREACIONES ARQUITECTÓNICAS

RhinoScripting Workshop :: Lima 2008 :: www.espaciosdigitales.org/lima/1

Veamos

_Sintaxis para estructurar funciones a través de la programación

_Creación de variables, que pueden ser fácilmente modificadas, y en cualquier momento, obteniendo diferentes resultados.

_Desarrollar algoritmos en base a funciones matemáticas (**trigonometría**), que pueden ser aplicadas a cualquier **calculo estructural**.

_Obtener información del usuario, logrando interacción entre máquina y arquitecto.

_Aprender pequeñas funciones que nos permiten averiguar , dónde y como se esta cometiendo un error.

Ej: "Comando **Rhino.Print**"

_Aplicación de la función "Random", estableciendo mínimos y máximos. (**Permite obtener distintas posibilidades a partir de una idea**).

_Funciones recursivas, funciones condicionales, extrusión, y Arrays.

_Insertar automáticamente figuras en puntos determinados, según función (Ej: **Creando reglas, como articulaciones entre los distintos vectores**)

_EN CONCLUSIÓN LA PROGRAMACIÓN NOS PERMITE TENER UNA VARIEDAD DE RESULTADOS, BASADOS EN PARÁMETROS DADOS POR EL PROGRAMADOR. LO POSITIVO ES QUE ESTOS PARAMETROS, SE ADAPTAN A CUALQUIER CIRCUNSTANCIA O MOMENTO, CON VARIACIONES MÍNIMAS Y RÁPIDAS.

AHORA... ¿QUE APRENDIMOS?

RhinoScripting Workshop :: Lima 2008 :: www.espaciosdigitales.org/lima/1