



Scripting como estrategia de diseño: una experiencia pedagógica

Ernesto Bueno

Aspirante a doctor en arquitectura por la Universitat Internacional de Catalunya, Barcelona, España.
ernestobueno@gmail.com

Carlos de la Barrera

Aspirante a doctor en arquitectura por la Universitat Politècnica de Catalunya, Barcelona, España.
cidelab@gmail.com

Resumen. Este artículo presenta nuestra visión del Scripting como técnica para ser incluida en el proceso de diseño arquitectónico. Esta visión no sólo parte desde nuestra experiencia como arquitectos investigadores, sino también desde la experiencia de un taller que hemos impartido recientemente. Afirmamos que la incursión de esta técnica beneficia al arquitecto en su proceso creativo, tanto por la automatización de procesos de modelado, como por la posibilidad de generación emergente de geometrías complejas pero controladas, respondiendo así a los nuevos paradigmas de la arquitectura y el diseño con procesos racionales. Se discuten los resultados de esta corta experiencia pedagógica con logros interesantes y la potencial trascendencia de la enseñanza de esta técnica en las disciplinas de diseño y arquitectura.

Palabras clave: estrategias pedagógicas, programación, taller de diseño, scripting

Abstract: This paper presents our vision of Scripting as a technique to be included in the process of architectural design. This vision not only shares our experience as research architects, but also from the experience of a workshop that we conducted recently. We affirm that the incursion of this technique benefits the architect in his creative process, both by automating modeling processes, and by making possibly the emergent generation of complex but controlled geometries, in response to the new paradigms of architecture and design with rational

processes. We discuss the results of this short educational experience with interesting achievements and the potential significance of teaching this technique in the architecture and design disciplines.

Keywords: design workshop, programming, scripting, teaching strategies

I. INTRODUCCIÓN

En la escena actual de la arquitectura está empezando a ser conocido el diseño emergente, apoyado en nuevos métodos y tecnologías, capaces de implementar ideas como: geometrías no estándar, customización, sistemas generativos, sistemas evolutivos, eficiencia estructural, acondicionamiento ambiental o sustentabilidad.

Una de esas técnicas es Scripting, con esta técnica se componen procedimientos como guiones de instrucciones para que una aplicación interactúe con el objeto que está siendo diseñado, aumentando de esta manera el alcance de dicha aplicación.

Es cada vez mayor la cantidad de diseños, resueltos a través de Scripting, que figuran en concursos internacionales y en nuevos proyectos arquitectónicos que son divulgados en las principales publicaciones de nuestra disciplina.

Así como para nosotros es imperativa la implementación de ésta técnica en nuestro ejercicio como arquitectos, consideramos que también es importante difundirla a través de cursos especializados, como una contribución a la formación de la presente generación de diseñadores.

Por estas razones implementamos un taller de diseño adaptado al entorno a través de Scripting en el Postgrado de Diseño, Entorno y Arquitectura de la ELISAVA, Escuela Superior de Diseño en Barcelona, España.

A. Antecedentes

Nuestros trabajos en el campo de Scripting nacen de la iniciativa de desarrollar estrategias emergentes de diseño como elemento de gran importancia para la realización de nuestras investigaciones doctorales. La mayoría de nuestro conocimiento en ésta área es autodidacta.

En cuanto a talleres de RhinoScript para arquitectos y diseñadores, nos anteceden varios impartidos en diferentes universidades europeas y norteamericanas, como los de la Universidad de Kassel por Schein y Kossmann (2002-2004) [1]; del MIT por Griffith y Snaveley (2006) [2]; o de la Universidad de Columbia por Andrasek y Bearak (2007) [3]; entre otros.

En el ámbito latinoamericano, se encuentran los talleres de la Universidad de Chile por Griffith, Snaveley y Cardoso (2006-2007); y de la Universidad Peruana de Ciencias Aplicadas por Griffith, Cardoso y Herrera (2008). Ambos casos coordinados por P. Herrera y documentados en su ponencia del SIGraDi 2007 [4].

El presente taller ha evolucionado de la experiencia de Carlos de la Barrera en talleres similares, como los impartidos en la Escuela Superior Artística de Porto [5], o en el Instituto de Arquitectura Avanzada de Cataluña [6].

B. Objetivos del taller

- Introducir al diseñador en el uso de la programación como estrategia de diseño.
- Complementar y ampliar conocimientos de modelado y diseño con Rhino a través de RhinoScript.
- Aportar eficiencia al diseño con la automatización de procesos permitiendo una mejor administración del tiempo.
- Permitir el manejo de geometrías complejas de una manera lógica logrando así resultados más coherentes.
- Fomentar la iniciativa y la creatividad en el uso de nuevas técnicas en el diseño.
- Contribuir a la enseñanza y divulgación de las tecnologías emergentes en el diseño y la arquitectura.

II. MATERIALES Y MÉTODOS

De todas las técnicas de Scripting en uso actualmente, nuestro desarrollo se hace en RhinoScript, debido a que Rhino es la aplicación de modelado con la que mejor nos manejamos (Rhino es desarrollado por McNeel, basado en OpenNURBS, que es una librería Open Source. RhinoScript es un lenguaje basado en VBScript). Además, Rhino es la aplicación con la que ya se introduce al alumno del postgrado en el modelado en 3D.

```
Sub FaceSurface()
Dim strSurface
Dim arrDomainU, arrDomainV
Dim ParamU, ParamV, arrPoint
Dim i, j
Dim SecU, SecV

SecU = 10
SecV = 10

strSurface = Rhino.GetObject("Selecciona una superficie")
If IsNull(strSurface) Then Exit Sub

arrDomainU = Rhino.SurfaceDomain(strSurface, 0)
arrDomainV = Rhino.SurfaceDomain(strSurface, 1)

For i = 0 To SecU
For j = 0 To SecV
ParamU = arrDomainU(0) + i * ((arrDomainU(1) - arrDomainU(0)) / SecU)
ParamV = arrDomainV(0) + j * ((arrDomainV(1) - arrDomainV(0)) / SecV)

arrPoint = Rhino.EvaluateSurface(strSurface, Array(ParamU, ParamV))
Call Rhino.AddPoint(arrPoint)
Next
Next
End Sub
```

Figura 1. Fragmento de la subrutina principal del script base desarrollado por C. de la Barrera.

El taller se comenzó con una breve explicación de los fundamentos de programación y de los principales conceptos del RhinoScript, tales como la estructura de un script, el manejo de datos con variables y los tipos de datos. Al mismo tiempo los alumnos fueron introducidos en nuestra manera de trabajo con Monkey Editor (un Plug-in para Scripting en Rhino 4 desarrollado por Rutten para McNeel, de libre distribución) y en el uso del archivo de ayuda de RhinoScript. Seguidamente, se les hizo escribir un script base, previamente preparado, con el cual se puede evaluar una superficie dada para construir sobre ella lo que sea necesario (Fig. 1). El proceso fue dado paso a paso, explicando cada uno de los comandos y métodos.

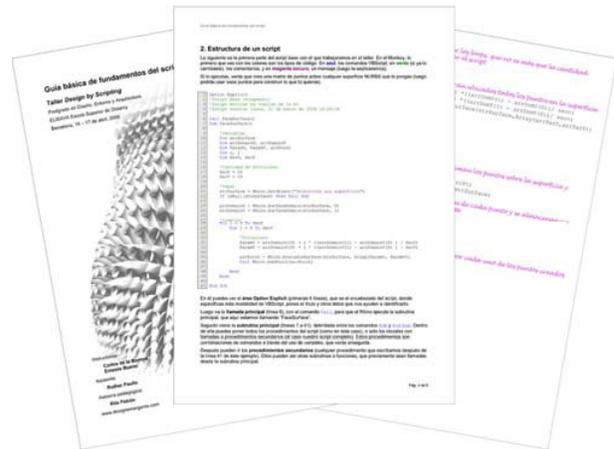


Figura 2. Guía redactada y entregada a los alumnos como material de apoyo para el taller, en el que se detalla el script base, al tiempo que se explican los conceptos utilizados.

Una vez escrito y entendido el script base, y entregado el material de apoyo (Fig. 2), se alentó a los alumnos a proponer un proyecto sobre el cual desarrollar un diseño arquitectónico basado en éste script.

Con nuestra supervisión y apoyo los participantes del taller generaron sus proyectos, presentándolos con imágenes y videos. (Fig. 3-9).



Figura 3. Columnas irregulares entre 3 superficies. Proyecto de Madeira Escamilla, Miguel Lozano, Maria Fernanda Cabrera, Mercé Labordena y Gloria Clavera.

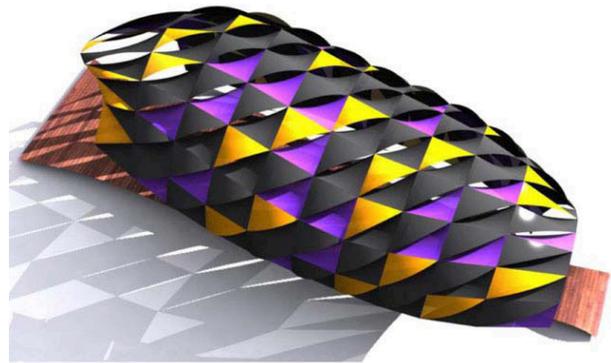


Figura 6. Cubierta de sacos triangulares. Proyecto de los alumnos: René Daniels, Daniela Martínez, Leila Pablo, Noel Luine y María Gabriela Dávila.

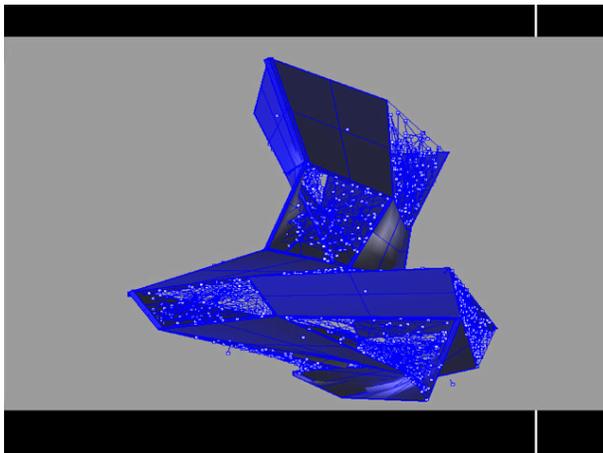


Figura 4. Patrones de crecimiento de objetos “tejidos” sobre superficies. Proyecto de Ginnette Gotti, Jorge Luis Sanchez, Cintia Torres, Ramón Cárdenas y Carlos Briceño.

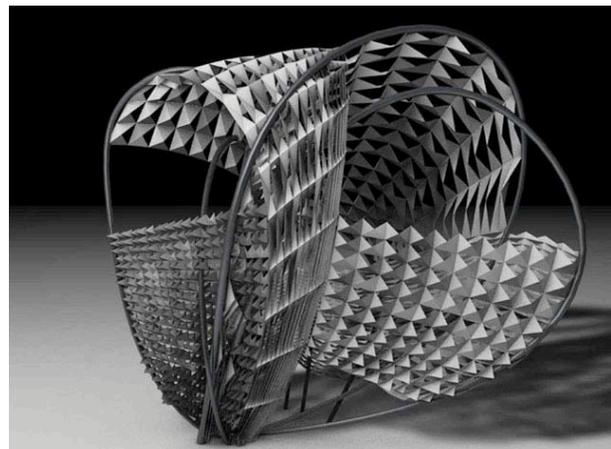


Figura 7. Cubiertas de celdas piramidales. Proyecto de Fanny Flores, Arturo Leal, Aylin Suárez, Daniel Gómez y Eva Ferradás.

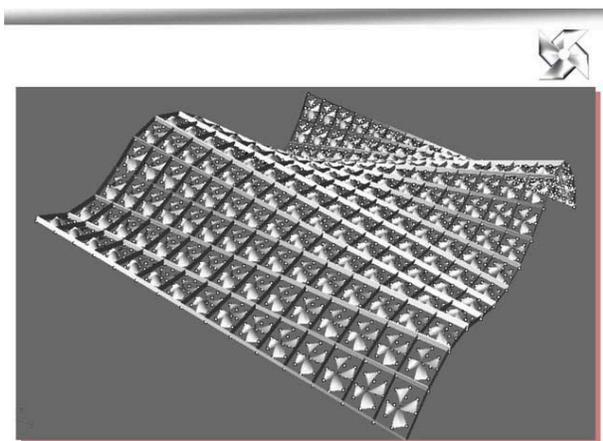


Figura 5. Proyecto Éxle: instalación de elementos móviles dentro de una superficie. Por Rita Rodrigues, Daniela Camargo, Africa Arreola, Feliz Leveque y Felipe Salazar.

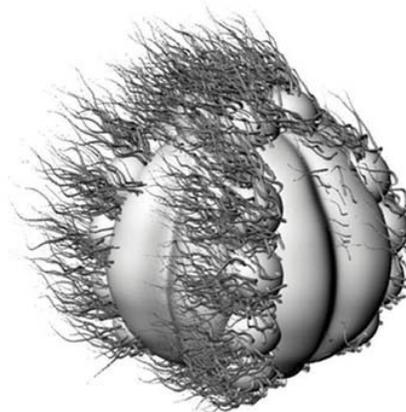


Figura 8. Crecimiento aleatorio de pelos. Proyecto de Camilo Molano, Cesarina Medina, Delia Arce, Ivan Herrera y Mariam Sadedine.

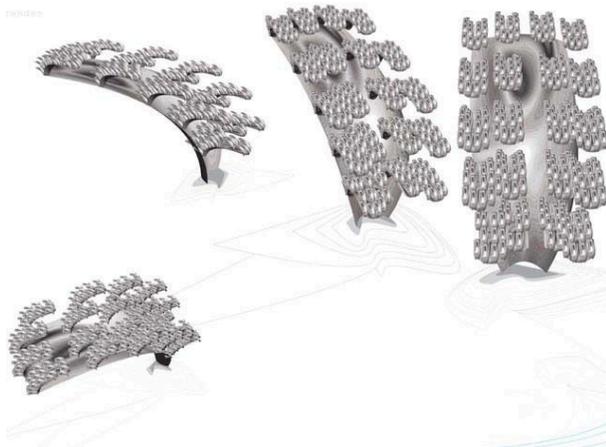


Figura 9. Crecimiento recursivo de superficies. Proyecto de González, María Cuadrado, Stella lo Sciuto, Fernanda Aiex Da Fonseca y Carlos Báez.

III. RESULTADOS Y DISCUSIONES

En el modelado en general, saber Scripting permite, en primer lugar, automatizar procesos, ahorrando un tiempo valioso. Con la tendencia a la complejidad que observamos en la arquitectura actualmente, automatizar procesos se torna una necesidad. Además, el Scripting da la oportunidad de lograr situaciones impredecibles a través del uso de la aleatoriedad en la programación, en caso de ser necesario. Pero sobre todo creemos que un alumno que maneje la lógica booleana inherente al Scripting puede tener más habilidad para solucionar problemas de diseño de una manera coherente.

La primera reacción negativa en los alumnos fue por la necesidad de usar las matemáticas. Es de nuestra opinión que, en general, a los estudiantes de diseño y arquitectura no les gusta los esfuerzos intelectuales que involucran cálculos aritméticos o algebraicos, por sencillos que sean. Observamos también algo de dificultad en entender el pensamiento booleano, pero en especial en entender los Arrays (variables que contienen un conjunto ordenado de datos, como las coordenadas de un punto, o una nube de puntos, etc.)

Con la disposición de tan sólo tres sesiones se logró que los alumnos manejaran fundamentos de programación y desarrollaran más a fondo el script, añadiendo sus propios procedimientos para llevar a cabo las tareas requeridas para la resolución de sus proyectos.

Nuestro principal inconveniente fue el corto tiempo con el que contamos para dar el taller, y los alumnos también lo manifestaron así. Por esta razón tuvimos que pasar por encima varios contenidos que nos hubiese gustado profundizar.

Sería beneficioso para los alumnos si contáramos con más tiempo en un próximo taller. De ser así, asignaríamos un ejercicio introductorio con el cual esclarecer los fundamentos de VBScript, lo que les permitiría un resultado más inmediato, e igualmente les proporcionaría mayor tiempo de ejercitación en el uso de la aplicación, transmitiendo una mayor seguridad en manejo de la técnica. De esta manera se facilitaría el aprendizaje del script base con el cual desarrollar los proyectos (que en nuestro taller abarcó toda la primera sesión).

Al final, los alumnos expresaron su entusiasmo e interés por la técnica a la que calificaron de potente e importante para el diseño arquitectónico. También comunicaron su satisfacción por el resultado obtenido en sus proyectos desarrollados durante el taller y su deseo de continuar aprendiendo Scripting.

IV. CONCLUSIONES

Los resultados del taller fueron muy satisfactorios, ya que los alumnos lograron ampliar conocimientos de modelado y diseño, a la vez que generar sus proyectos utilizando RhinoScript. El Scripting demostró ser útil para el proceso de diseño, y por consiguiente, para la enseñanza de la arquitectura. Tanto por su apoyo en el desarrollo de proyectos, automatizando procesos y optimizando tiempos, como por su impulso en el pensamiento lógico, creemos que la inclusión de esta técnica en la enseñanza del diseño es beneficiosa para el estudiante de arquitectura y puede tener una repercusión de gran importancia para la disciplina. De ahí que podría ser atrayente para las escuelas de arquitectura y de diseño, y se interesen en el desarrollo de esta técnica y la integren en sus programas curriculares.

RECONOCIMIENTOS

Los autores agradecen a Ignasi Pérez Arnal, director del Postgrado de Diseño, Entorno y Arquitectura, por la oportunidad de impartir el taller. También agradecen a Ruther Paullo, asistente, y a Rita Patron y Euridys Wills, asesoras pedagógicas.

REFERENCIAS

- [1] Schein, Markus. (2005). Advanced CAD: Rhino Scripting. Disponible en: http://www.uni-kassel.de/~digitalp/epool/html/rhino_scripting/web/rhinoscripting_start.html. Acceso en: 26 de marzo de 2008.
- [2] Snavelly, John; Griffith, Kenfield. (2006). Computational Design Solutions. Disponible en: <http://www.mit.edu/~snavelly/rhinoscripting/>. Acceso en: 5 de junio de 2008.
- [3] Andrasko, Alisa; Bearak, Mark. (2007). Material Potency Spring '07. Disponible en: <http://metaware.wetpaint.com/page/Material+Potency+Spring+07>. Acceso en: 9 de julio de 2008.
- [4] Herrera, Pablo. (2007). Solución de problemas relacionados al diseño de superficies complejas: Experiencia de programación en la educación del arquitecto. 11º Congreso Iberoamericano de Gráfica Digital, México. pp. 97-101.
- [5] de la Barrera, Carlos. (2008). "RhinoScripting Workshop at Esap, Oporto". En: G.D.E.S. Disponible en: <http://www.designemergente.org/rhinoscripting/rhinoscripting-workshop-at-esap-oporto/>. Acceso en: 13 de abril de 2008.
- [6] de la Barrera, Carlos; Fraguada, Luis. (2008). "S6: Scripting Seminar". Disponible en: IaaC Blog. <http://www.iaacblog.com/scripting/?p=3>. Acceso en: 12 de mayo de 2008.

Ernesto Bueno es arquitecto, tiene un Máster en Arquitecturas Genéticas por la ESARQ – UIC, donde actualmente cursa su doctorado en la misma línea de investigación. Ha sido asistente de profesor en asignaturas de CAD e infografía en Arquitectura, y ha trabajado como arquitecto colaborador en algunos despachos de arquitectura en Barcelona. Ha sido ponente y comité organizador de congresos de informática en arquitectura, como el SIGraDi. También ha sido coautor de proyectos ganadores en concursos de arquitectura. Actualmente investiga acerca de las estructuras de ramificaciones, y su desarrollo por medio de scripts. Sitio Web: <http://ernestobueno.googlepages.com/>. Correo electrónico: ernestobueno@gmail.com

Carlos de la Barrera es arquitecto, tiene un Máster en Arquitecturas Genéticas por la ESARQ – UIC y cursó el programa de Técnico en Programación de lenguaje C++ y Visual Basic, del Instituto Superior de Programación (CYPSA). Actualmente desarrolla su doctorado Algoritmos Genéticos como Estrategia para el Diseño de Arquitectura, con Beca de la Fundación Rafael Escolá. El estudio explora la generación y el crecimiento de superficies por medio de algoritmos computacionales basados en estrategias de la evolución biológica. Imparte clases en la Escola Artística Superior do Porto (ESAP), y en el Institut d'Arquitectura Avançada de Catalunya (IaaC). Sitio Web: <http://www.designemergente.org/>. Correo electrónico: cidelab@gmail.com