



Archfarm

Fascículos aperiódicos de arquitectura

NÚMERO 7 † JULIO 2005

www.archfarm.org

HARDSPACE, SOFTSPACE:

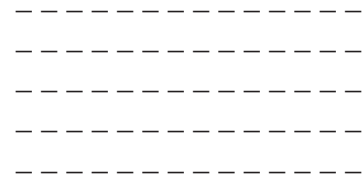
ARQUITECTURA DE CÓDIGO ABIERTO

USMAN HAQUE

Nuestro entorno construido, con su impacto diario en la gente y su constante transformación con el uso y re-uso, es un proyecto diseñado colectivamente que incorpora considerables diferencias y algunas veces lógicas contrapuestas.

Las cuestiones que surgen de las diferentes perspectivas y acercamientos de la gente darán consecuencias significativas en la manera en que la arquitectura en general va a evolucionar en el siglo XXI.

La terminología informática ha tomado mucho prestado de la disciplina arquitectónica; aquí, vamos a recoger algunas analogías desde el mundo informático para sugerir caminos hacia los que la arquitectura puede evolucionar.



Tradicionalmente se ha pensado en la arquitectura como *hardware*: los muros, cubiertas, y suelos estáticos que nos envuelven. Una aproximación alternativa es pensarla como *software*: los sonidos, dinámicos y efímeros, olores, temperaturas e incluso las ondas de radio que nos rodean. También podríamos considerar las infraestructuras sociales que apuntalan nuestros espacios diseñados. Llevando esta analogía más lejos, podemos pensar la arquitectura como un *sistema operativo*, dentro del cual la gente pueda crear sus propios programas para la interacción espacial.

Portada
cortesía de
David Cuesta,
david@mail.
themcdc.com

Traducción:
Archfarm

Página web
del autor:
www.haque.
co.uk

El diseño arquitectónico que enfatiza el *softspace* por encima del *hardspace* es un poco como el diseño de *software* en lugar del diseño de *hardware* en términos informáticos, donde *hardware* se refiere a la máquina física y *software* a los programas que la hacen funcionar. En el contexto arquitectónico, la tecnología se usa para provocar interacciones entre la gente y entre ésta y el espacio físico. Si el *softspace* anima a la gente a convertirse en intérpretes de su propio entorno, entonces el *hardspace* proporciona una estructura que invita a estas interacciones. La idea de un *sistema operativo arquitectónico* yace en el diseño de los sistemas que integren a ambos, *hardspace* y *softspace*. Un modelo de sistema operativo que es particularmente relevante para la arquitectura —dando por supuesto que el diseño del espacio es siempre un proceso colaborativo— es un *sistema de código abierto*.

Todo el mundo es diseñador de espacio y todos usamos nuestro espacio e interfaces de maneras

diferentes. Ponemos posters en las paredes, las pintamos de azul o naranja, colocamos los muebles en las habitaciones, hacemos el amor en la cocina, usamos “dormitorios” como “oficinas”, cantamos ópera en la ducha, vaporizamos fragancias en nuestros baños y usamos las escaleras para discusiones, juegos o romances. Mientras tanto, cada vez es más posible emprender la construcción o mejora de nuestras casas sin la necesidad de los servicios de un arquitecto. Aun así, la mayoría de la gente no se ve capacitada para *diseñar*.

Incluso en un entorno diseñado por un arquitecto, los desarrollos tecnológicos llevan a cuestionar el verdadero rol del arquitecto, porque los mecanismos interactivos entre usuario y entorno permiten a la gente coger una posición principal en la configuración y diseño de su propio espacio. La representación más sencilla de estos mecanismos es el termostato, que regula la temperatura de acuerdo a nuestro requerimientos; más recien-

temente sistemas que permiten cambiar el color, textura, configuración y transparencia de muros, y otros que sugieren el proceso circular de *conversación* con el entorno, han dejado claro que los arquitectos están perdiendo la prioridad en la definición de los límites de los deseos y movimientos de la gente. ¿Qué es, entonces, lo que hace el arquitecto?

Si el arquitecto diseña sistemas de interacción, entonces, la producción de la arquitectura, que existe sólo en el momento de usarse, es puesta en las manos del usuario final. El diseño arquitectónico, el coreógrafo de sensaciones, puede proporcionar metaprogramas con los que la gente construya sus propios programas. En informática, un *sistema operativo* —como Unix, Windows o Mac OS— es el software que hace funcionar un ordenador en su nivel primario, y que actúa como plataforma sobre la que ejecutar otros programas. Extendiendo la analogía a la arquitectura, un *sistema operativo espacial* propor-

ciona estructuras para fomentar multitud de programas arquitectónicos. Según esta idea, la gente es la diseñadora de sus propios espacios y los arquitectos simplemente diseñan los metasisistemas.

Los sistemas de interacción eliminan distinciones entre audiencia e intérpretes, usuarios y diseñadores, ocupantes y arquitectos, y abren posibilidades creativas para el diseño de espacios, eventos y situaciones. También aumentan los desafíos para el papel social de los diseñadores de proveer metasisistemas que fomenten la creatividad individual y alienten a la gente a coreografiar sus propios programas espaciales, a diseñar sus propios espacios e inventar sus propias lógicas. El dilema es diseñar sistemas operativos que promuevan la creatividad sin añadir más capas reguladas de control.

Código abierto se refiere, en el universo del software, a un tipo de código fuente —con el que el software es diseñado y construido— que es accesible a todo el mundo,

que es libremente distribuido mientras permanezca abierto, que permite modificaciones y derivaciones —que a la vez han de ser abiertas—, que es no discriminatorio, en el que es posible introducir parches sin perturbar la integridad del trabajo total, y que es tecnológicamente neutral. De manera similar, una *arquitectura de código abierto* necesita una estructura en la cual la distinción entre *aquellos que diseñan* y *aquellos que usan* es sustituida por un sistema participativo que alienta un proyecto destinado a estar constantemente *parcheado* o *modificado*.

Una arquitectura muy cercana a las visiones que el artista, arquitecto y situacionista holandés Constan tuvo en su proyecto *New Babylon*. En esta enorme exploración Constan asumió que todo el mundo es un artista en el diseño y construcción de sus espacios, eventos y vidas. Su proyecto proponía una estructura mundial en constante construcción y reconstrucción por sus habitantes. Una

estructura que cambiara en toda su extensión según diferentes grupos de gente la alteraran contribuyendo de diferentes maneras. Constan redujo la distancia entre las prácticas del arte y de la arquitectura y señaló las conexiones entre el deleite del arte y el de la arquitectura. Sin embargo, el proyecto plantea una importante cuestión: si todo el mundo es artista y todo es arte —léase arquitectura— ¿no implica eso que, sin distinciones, nada es arte?

El movimiento de código abierto en software nos da algunas pistas sobre cómo resolver este rompecabezas. Nos ofrece un sistema que, en teoría, es suficientemente abierto para que cualquiera pueda unirse, pero no ha significado en la práctica que todo el mundo se esté haciendo programador. Están los que usan el sistema por el reto de construir nuevo código y los que disfrutan la cultura de código abierto sin necesidad de contribuir al proceso de construcción. De manera similar, aplicar el código

abierto a la arquitectura sugiere un proyecto democrático colaborativo que exista en el tiempo tanto como en el espacio: una arquitectura que es creada por gente mediante su uso, como una *performance*, una conversación, un *bodystorm* que se forma a lo largo de la vida del sistema arquitectónico—un edificio u otro contexto arquitectónico—.

El teórico y diseñador industrial Anthony Dune, hablando sobre el diseño de lo que él llama *objetos post-óptimos*¹ dice: «El desafío más difícil para los diseñadores de objetos electrónicos radica ahora no en la funcionalidad técnica y semiótica, donde se alcanzan ya niveles óptimos, sino en los ámbitos de la metafísica, poesía y estética, que se han investigado poco.»² Si asumimos que estos sistemas en arquitectura pueden resolver los requerimientos prácticos y funcionales de los espacios construidos, entonces la belleza en el diseño viene de la poesía de aquellos que la usan, implementan y rehacen. Un sistema que permitiera a la gente crear

por sí mismos sus propios espacios y construir colectivamente un espacio social, que sería más conceptualmente abierto.

Hay varias características clave que definen una arquitectura de código abierto:

1. DISEÑADORES – PARTICIPANTES. Quienes participan en el sistema son también quienes lo diseñan.

2. UN SISTEMA DE CONTROL del que uno se hace parte para expandir dicha estructura: un ejemplo son los videojuegos de ordenador, que proporcionan módulos que permiten al usuario crear sus propios escenarios, a veces sorprendentemente diferentes.

3. COREOGRAFÍAS PARA LA APERTURA. Instrucciones de grupo interpretadas y modificadas como sea necesario por los participantes, individual o colectivamente. Para empezar es necesario establecer unos límites que fomenten la creatividad. Esto no quiere decir que éstos no puedan ser rebasados. Se establecen como referencias y no para predefinir barreras.

4. REAPROPIACIÓN. Los espacios, objetos o acciones existentes sirven de combustible y de catalizadores para la creatividad.

5. CAPACIDAD PARA COMPARTIR-LOS PROBLEMAS DE DISEÑO. Cada persona tiene diferentes aptitudes y a menudo un problema requiere una solución que sólo puede proporcionar otro. Un ejemplo basado en la web, *lazysweb.org*, muestra cómo no es importante que todo el mundo tenga las capacidades técnicas para llevar un modelo de producción de código abierto.

En un futuro inmediato la arquitectura de código abierto necesitaría dar dos pasos diferentes: El primero sería desarrollar infraestructuras que permitan a los diseñadores *no profesionales* participar más activamente en los procesos de diseño y construcción. En cierto sentido esto ya está ocurriendo, como muestra la tendencia de la autoconstrucción³. Sin embargo, los arquitectos profesionales pueden hacer mucho más para facilitar

1

«Objetos diseñados una vez que la funcionalidad se da ya por supuesta»

2

Anthony Dune. *Hertzian tales: electronic products, aesthetic experience and critical design*. Royal College of Art, Londres, 1999.

3

www.readymademag.com

la transición. Pragmáticamente, los arquitectos tienen la oportunidad en este punto de participar en las conversaciones que hacen posible y fomentan el diseño de buenos edificios y la práctica colaborativa. Esto puede ocurrir a un nivel práctico expandiendo las prácticas participativas en la industria; quizá también en el discurso teórico, donde las ideas y conceptos que esconde la arquitectura necesitan ser abiertos.

El segundo sería aplicar el conocimiento del diseño de espacios a la formulación de una estructura con la que otra gente pudiese, deliberadamente, diseñar espacio. En esta cualidad, los arquitectos deberían alentar el reconocimiento de la distinción entre *buen* diseño y *mal* diseño, si se puede decir que exista. Otra vez, este paso puede ser localizado prosaicamente en la práctica actual de la industria. Sin embargo es también necesario expandir el discurso teórico sobre cómo *diseñar* *diseño*. Un sistema operativo espacial reconoce que todo el mun-

do es ya un diseñador: sería vital en este punto asegurar que los arquitectos no se conviertan en otro metasistema que “objetivamente” controle el proceso desde arriba. Más que dirigir, los arquitectos deberían sobretodo transformarse en facilitadores o cooperantes.

El rol de la arquitectura experimenta cambios considerables porque la propia gente interpreta, diseña, reusa y se apropia de su propio espacio según sus propias lógicas. Una verdadera arquitectura de código abierto no existe sin gente que la habite, que la ocupe, la perciba, que interactúe o converse con ella. Los espacios resultantes no solamente permiten a la gente desarrollar sus propias formas de interacción, sino que realmente se enriquecen con ello. Al volverse las personas arquitectos de su propio espacio a través de su uso, o desarrolladores de sus propias interfaces, las palabras *arquitectura* e *interface* dejan de ser nombres para convertirse en verbos. Tal arquitectura es explícitamente dinámica,

un cambio que abre una riqueza de posibilidades poéticas para los diseñadores de espacio de código abierto.

Sabemos que la arquitectura es política. Y sabemos también que la propia gente hace arquitectura al usarla. El reto ahora es equilibrar las diferencias entre habilidades técnicas, acceso tecnológico y deseo de autosuficiencia que diferente gente tiene, con el objetivo de producir un espacio democrático viable, en todos los sentidos.

¿Son todos los sistemas de arquitectura metasistemas de control? El código abierto y procesos similares de diseño en colaboración sugieren que hay otros caminos hacia delante.

