



Tipografía orgánica

Daniel R. Valero

Vaya por delante que el título del artículo se lo debo al gran Gerard Unger; tras una conferencia en el tercer congreso internacional de tipografía (Valencia, 2008) le pregunté si utilizaba o, al menos, si había probado sistemas paramétricos de diseño de tipos como METAFONT. Fue entonces cuando oí por primera vez la distinción entre tipografía *sintética* y *orgánica*, y la impresión que me produjo fue la de una genialidad de una sencillez abrumadora, de esas que distinguen a los grandes maestros.

Aquí el término *sintética* no tiene connotaciones negativas, hace referencia a una manera de fabricar letras descriptiva, artificial e industrial, basada en secuencias de vectores que crean/dibujan los límites entre grafismo y contragrafismo. No se trata más que de una continuación de la idea subyacente en los arquitectos renacentistas, la Romain do Roi y el pantógrafo. Se trata de un sistema poco limitador: en esencia, permite colocar los puntos que definen los perfiles en cualquier posición dentro de un espacio gráfico bastante amplio — un sistema de coordenadas de 1.000 x 1.000 unidades en el formato PostScript o de 2.048 x 2.048 en el formato TrueType— y, por lo tanto, capaz de generar un número prácticamente infinito de formas diferentes.¹

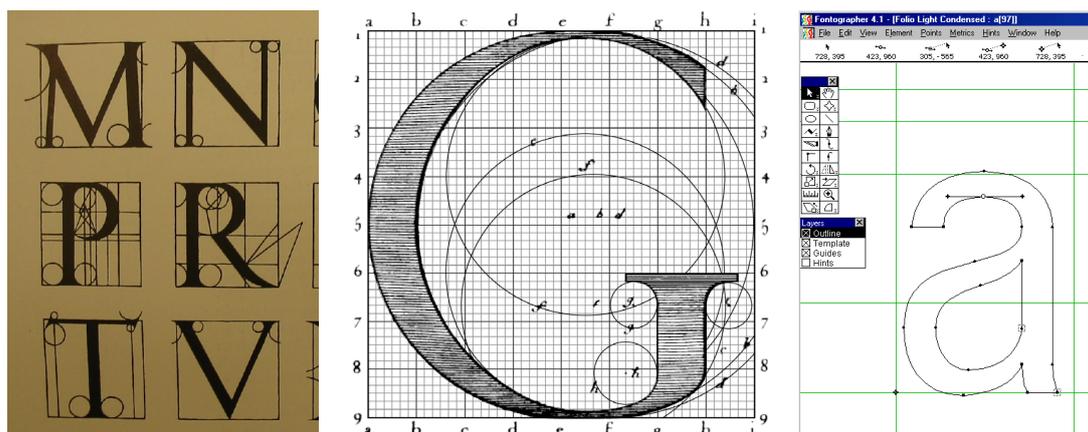


Fig. 1: Representación geométrica de Francesco Torniello (1517), Romain do Roi (1702), Fontographer 4.1. (1998).

La consecuencia lógica de esta técnica es un gran despliegue de tipos diferentes, desde la reinterpretación de los grandes clásicos, la recuperación de los distintos modelos caligráficos cultos y vernáculos, hasta la creación de nuevas formas híbridas, pastiches, deconstrucciones y demás elementos postmoder-

¹ Sería prácticamente imposible calcular el número exacto de tipografías que existen hoy por hoy por varias razones: cualquiera puede fabricar una nueva, compartirla en la red o crear una fundición para su distribución comercial sin grandes conocimientos previos y en cuestión de horas. Como ya es sabido, en internet la cantidad de información acumulada en unos pocos años excede con mucho todo lo escrito desde que se inventó la escritura, hace unos cinco mil años.

nos. Cada cultura crea, recupera o amplía la tecnología necesaria para cumplir sus propósitos, con lo que la tipografía sintética encaja perfectamente en una sociedad totalizadora como la nuestra, donde conviven rasgos primitivos, medievales o neobarrocos.

BEGOÑITA *Begoñita*
Begoñita Begoñita
 Begoñita Begoñita

Fig. 2: Nunca antes una misma tecnología ha sido capaz de crear formas tan diferentes, en este ejemplo tenemos una capital imperial esculpida (Trajan), una gótica caligráfica (Fette Fraktur), una tipografía clásica (Garamond), una letra vernácula (Bello), una tipografía *grunge* (Trixie) y un pastiche (Dead History).

El término sintético, referido a un producto, significa “obtenido por procedimientos industriales, generalmente una síntesis química, que reproduce la composición y propiedades de algunos cuerpos naturales.” (DRAE) Aquí la primera parte de la definición no se cumple *sensu stricto*, pues no podemos considerar la computadora como un procedimiento industrial, si acaso una herramienta tecnológicamente avanzada que opera conceptualmente de manera artesanal pero que todavía imita la estética industrial que le precede (Rodríguez, 2004). Lo interesante de la definición es la segunda parte: *reproducir la composición y propiedades de algunos cuerpos naturales*; la tipografía sintética, por puro determinismo técnico, tiende a reproducir² la composición y propiedades de la tipografía en plomo y la caligrafía, de hecho está planteada para representar cualquier tipo de forma. Eso la convierte en el estándar de facto de la industria informática, primero con el programa IKARUS en la década de los setenta, el formato PostScript y Fontographer en los ochenta, TrueType en los noventa y OpenType junto a FontLab en nuestra década.³ Por último, cabe destacar que la tipografía sintética es indirecta: las astas de las letras no se tratan como algo independiente, sólo se intuyen del dibujo general. ¿Puede haber algo más artificial?

En el extremo contrario, hay otra manera de fabricar letras, explicativa, natural y artesana. En este método se trata de crear escenarios a partir de elementos predefinidos, que pueden ser módulos o preceptos, reglas o comportamientos

² El concepto *reproducir* en sí mismo nos sirve como síntoma de una disfunción conceptual típica del entorno digital: la reproducción es natural a los procesos analógicos, mientras que la recreación lo es a lo digital. La tecnología ha cambiado más rápido que nuestras estructuras cerebrales, y tendemos a aplicar procesos analógicos en un sistema donde no son naturales, lógicos, incluso éticos.

³ No debemos olvidar a otros programas como FontForge, que es de código abierto, y otros formatos vectoriales como las espirales, que están empezando a plantearse como la alternativa de futuro en programas como InkScape, también GNU.

de ciertos elementos pre-existentes. Aquí los glifos y sus astas no se definen, se construyen de manera directa por combinación o por secuencia, por eso nos parece un sistema natural.⁴ De hecho, está emparentada conceptualmente con la quirografía y la caligrafía, con las que comparte limitaciones; sin embargo, éstas nunca han sido obstáculo para la creatividad, más al contrario, un estímulo, un acicate para crear reglas nuevas, parametros diferentes o simplemente, para buscar nuevos límites a cada tecnología.

Lo interesante es que la distinción entre tipografía sintética y orgánica sólo es posible con la aparición de la computadora; antes de existir ordenadores, sólo se podía hablar de tipografía sintética (recordemos: *producto obtenido por procedimientos industriales...*). Lo digital abre el abanico de posibilidades al alcance de los primeros investigadores que perseguían el desarrollo de sistemas de diseño de letras por computador.⁵

2 PIONEROS DE LA TIPOGRAFÍA ORGÁNICA

La primera aproximación al problema vino desde las matemáticas y la caligrafía. En 1967, H. W. Mergler y P. M. Vargo crearon ITSLF (InTeraCTive Synthesizer of LetterForms) influidos por las teorías de Fredrick Goudy. Fue el primer sistema paramétrico, aunque algo rudo, de la historia de la tipografía digital (Mergler y Vargo, 1976). En 1978, David Kindersley y Neil Wiseman crearon ELF, un sistema interactivo de dibujo, con tableta gráfica y lápiz óptico, capaz de dibujar trazos caligráficos de diferentes grosores, gobernados por parámetros (Kindersley y Wiseman 1979).

La idea general de estos sistemas consiste en crear una serie de parámetros que contengan información sobre las características de las letras. Estos parámetros son recreaciones verbales de los conocimientos de un diseñador de tipos, dentro de un rango determinado; el usuario puede cambiar el valor del parámetro y el sistema dibuja, mediante vectores o mapas de bit, la letra en cuestión.

El número de parámetros depende sólo de la imaginación del programador, las posibilidades son múltiples, y dejan abiertas posibilidades inéditas, como la casualidad, el contexto, etc.

2.1 El sistema Coueignoux

En la misma década mágica de los setenta en la que se publicó el trabajo del Dr. Hershey, Philippe Coueignoux (Coueignoux, 1975) creó un lenguaje llamado CSD (*Character Simulated Design*, o diseño simulado de caracteres) usando un conjunto de primitivas o módulos (ver figura 3) que, combinados entre sí mediante reglas y parámetros, construyen las diferentes letras y caracteres del

4 La mayoría de los sistemas escritos —salvo el Braille y el Morse— creados por el ser humano se componen de trazos, lo que determina la forma de sus glifos; quizás por esa razón dibujar perfiles en lugar de escribir trazos o astas de letras nos parece artificial.

5 Cuando hablamos de sistemas, hay que descartar el método empleado en los primeros años de tipografía digital, cuando las letras se dibujaban punto a punto, creándose un mapa de bits para cada cuerpo en cada resolución. Aquello era una locura, tanto por tiempo de dibujo como por problemas de almacenamiento.

alfabeto romano, controlando la proporción y la disposición de los módulos. Ghosh y Bigelow probaron una idea similar en 1983, pero el concepto no es nuevo en absoluto, pues sólomente hay que echar un vistazo al libro de Anduaga y Garmendia titulado *Arte de escribir por reglas y sin muestras*, en el que se explica cómo escribir una bastarda, letra genuinamente española⁶ por medio de tres principios, la *i*, la *r* y la *c* de caja baja (ver imagen). Y todo esto en 1781. El sistema no tuvo éxito comercial a pesar de ser, como comenta Peter Karow (Karow 1989), el método más lógico de rastrear fuentes digitales, pues el RIP sólo tendría que barrer relativamente pocos elementos, colocar juntos esos elementos y generar los mapas de bit de los caracteres a velocidades enormes (recordemos que la velocidad era un verdadero problema antes de la década de los noventa). Recientemente se ha hecho varias relecturas de este sistema, lo que demuestra que fue una persona lúcida y avanzada a su tiempo.

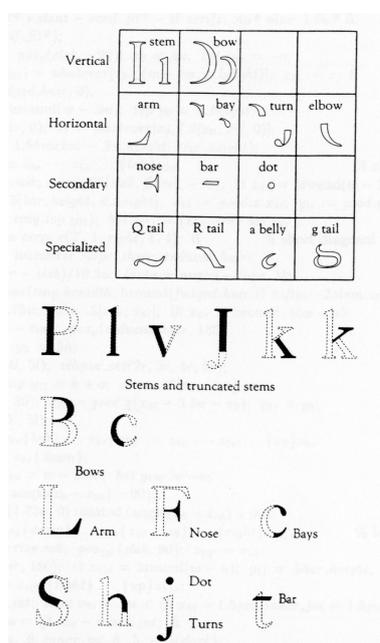


Fig. 3: Sistema Coueignoux

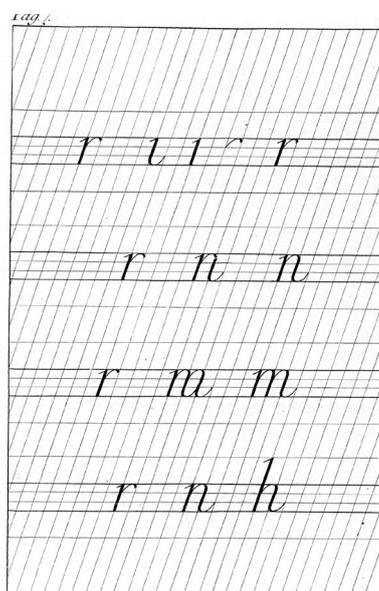


Fig. 4: Lámina del “Arte de escribir”, de Anduaga y Garmendia

2.2 Metafont

Mientras la tipografía sintética tomaba forma —IKARUS fue terminado en 1975 y PostScript en 1984— y tomaba posiciones hegemónicas en el mercado tipográfico, un profesor de la Universidad de Stanford llamado Donald E. Knuth desarrolló entre 1978 y 1988 METAFONT, un sistema de creación alfabética para su lenguaje TEX, un lenguaje descriptor de página similar al postscript, aunque más complejo y orientado exclusivamente a la composición de texto.⁷ Es

⁶ Cursus correspondiente a la cursiva, ductus flexible como en la letra inglesa.

⁷ Actualmente se utiliza mucho en círculos científicos, quienes componen sus textos en sistemas basados en TEX, como LaTeX, ConTEX o AMS-TEX.

tan completo que es el único sistema capaz de crear las complejas fórmulas y ecuaciones habituales en las ciencias exactas, pero su precisión es tal que se está usando como paradigma en otras aplicaciones comerciales, como Adobe InDesign.

Por lo tanto, METAFONT nació por una necesidad muy clara: la de surtir a TEX de los caracteres necesarios para la composición. Pero en lugar de crear una única descripción alfabética, comunmente llamada fuente, Knuth prefirió crear un sistema elástico, una meta-fuente,⁸ capaz de generar cualquier forma posible.⁹ METAFONT funciona desplazando un vector a través de una trayectoria, a la manera de un calígrafo —de hecho colaboró con Hermann Zapf en algunos de sus alfabetos— creando numerosos parámetros que podían ser alterados a voluntad del usuario (por ejemplo, la longitud de un corchete). Luego se generaba un bitmap para las resoluciones más comunes y ese mapa de bits era llamado por el documento en TEX que describía su posición, tamaño, etc.

La clave de METAFONT es que sus programas sugieren, declaran (*declarative*) más que forzar o ser imperativos (*imperative*), es decir, no hace falta dibujar todos los detalles de una forma gráfica, sólo especificar los componentes esenciales y las relaciones entre ellos; la computadora hace el trabajo sucio, gracias a un entrenamiento preciso, cierta inteligencia adquirida y que se encarga de tareas repetitivas, dejando al diseñador la mente libre para centrarse en otros aspectos.

El sistema no tiene interfaz gráfica de usuario (GUI); cada programador debe escribir las coordenadas de la trayectoria y los parámetros de cada forma. El resultado sólo es visible al final, cuando se le pide al sistema un mapa de bits adecuado a cierta resolución.¹⁰ Infinidad de variaciones pueden ser aplicadas a cada carácter. Los parámetros de METAFONT creados por el profesor Knuth son 62 (Knuth 1986).

Todos estos parámetros permiten crear un sólo alfabeto con diferentes aspectos (variantes de la familia gráfica o estilística, cuerpos, etc.) como en el caso de una fuente Multiple Master, pero con infinidad de posibilidades. Cada carácter es un programa independiente.

El grupo de trabajo de Knuth creó un alfabeto de libre distribución llamado Computern Modern, basado en el tipo Monotype Modern 8a. Gracias a los 62 parámetros se crearon variantes de la familia gráfica y estilística, y es la fuente en formato METAFONT más usada por los usuarios de TEX.

De hecho, una de las grandes ventajas de TEX y METAFONT es la gran comunidad de programadores trabajando para ampliar y mejorar el sistema, creando macros o herramientas externas que mejoran el rendimiento del programa, la conversión a formatos vectoriales, etc.

8 No debemos confundir una meta-fuente con una familia tipográfica serial, que contiene variantes de diferentes estilos, generalmente un palo seco, una romana y algún paso intermedio.

9 Al menos eso creían, cuando en realidad las posibilidades están restringidas por el número de parámetros.

10 Este es el gran defecto de METAFONT. Como explica Richard Kinch, creador de MetaFog (Kinch 1998) debería producir perfiles, no mapas de bit.

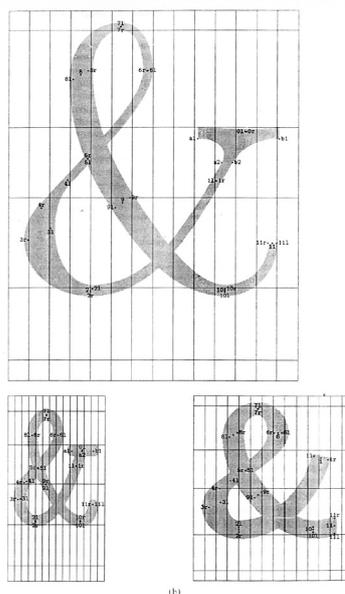


Fig. 5: Ligadura “et”, también conocida como ampersand, en formato METAFONT. Los tres caracteres son el mismo programa, sólo cambian unos cuantos parámetros.

3 LOS AÑOS NOVENTA: LA ORIENTACIÓN HACIA EL USUARIO

En la década de los noventa, mientras se debatía la guerra tipográfica —*Font Wars*— surgieron varias empresas independientes, pequeñas, con ilusión y gente formada en las grandes corporaciones, que desafiaron al mercado con pequeñas —y baratas— aplicaciones que acercaban al usuario al mundo de la tipografía digital paramétrica más allá de las *MultipleMaster*.

Hablamos en pasado porque desgraciadamente —y afortunadamente para las grandes multinacionales— estas aplicaciones han desaparecido. Fueron compradas por aquellos que se sintieron amenazados por ellas. Y es que como se apuntaba antes, la tipografía paramétrica no sólo acercaría la letra a todo el mundo, haciéndola más universal si cabe, sino que haría temblar los cimientos del mercado. En aquella época se pensaba que nadie querría comprar ya tipos si puede hacerse en un instante; bueno, casi nadie, y los que necesitaran comprarla no pagarían tanto pues los creadores no necesitarían tanto tiempo y esfuerzo en su fabricación.

Como sucedió en otros sectores del mercado informático, la sociedad aún no estaba preparada para los cambios que se atisbaban en aquellos años; las mentes preclaras que fueron capaces de anticiparse a la demanda no supieron ver el verdadero mercado y no tuvieron a su disposición los medios necesarios, como ordenadores baratos y potentes capaces de hacer los cálculos, una red rápida y usada por todo el planeta, etc. Como se comienza a intuir, es en la década actual cuando se pueden alcanzar resultados verdaderamente interesantes.

3.1 *Ares Font Chameleon*

Ares Software Corp., una compañía fundada en 1990 por tres antiguos miem-

bros de la plantilla de System Integrators Inc.,¹¹ Larry Applegate, Ernie Brock y Robin Henson, desarrolló una serie de herramientas para la manipulación de fundiciones digitales. Primero FontMonger y Font Chameleon, seguidas de FontMinder, FontFiddler y FontHopper.

La compañía tuvo bastante éxito entre los años 90 y 97, gracias a su Font Chameleon, que funcionaba creando nuevos tipos a partir de modelos preexistentes, gracias a parámetros ajustables por el usuario mediante barras (ver fig. 42), de un modo bastante intuitivo, pues se puede observar el cambio en tiempo real mientras se deslizan los controles de las barras.

Utilizaba una tecnología patentada llamada *difference descriptor*, que podríamos traducir como descriptor de diferencias. El programa interpolaba entre los extremos de cada descriptor, pero nunca entre ellos. El resultado era almacenado como un archivo descriptor de 5 Kb y se generaba una fundición PostScript o TrueType para Mac o Win perfectamente instruida para pantalla.

Se lanzaron tres versiones en apenas dos años, entre el 93 y el 94. Adobe compró la compañía en 1996, y el 6 de junio el desarrollo de sus productos fue paralizado. Poco tiempo después, toda huella de estas aplicaciones fue borrada del mapa, incluso en internet, no queda ni rastro de Chameleon, salvo peticiones de usuarios y comentarios sobre cómo funciona (Fabian 2000).



Fig. 6: Captura de pantalla del Ares Chameleon

3.2. HP Infont

A principios de los noventa, Hewlett Packard se había quedado atrás en la competición por el mercado de la impresión, y había tenido que claudicar adquiriendo licencias PostScript para sus impresoras, pese a haber intentado imo-

¹¹ Compañía que no sobrevivió al dominio aplastante de Adobe durante los años ochenta, para la cual crearon FontStudio, una herramienta de creación de fuentes vendida a Letraset. Es sorprendente que en todo el manual de usuario —436 páginas— no se haga mención a la contribución de sus autores al desarrollo del programa.

ner su propio lenguaje descriptor de página PCL sin éxito.

El 21 de diciembre de 1995 adquirió ElseWare Co., una empresa fundada en 1990 por Ben Bauermeister y Clyde McQueen, antaño empleados de Aldus, especializada en software para impresión y publicaciones.

ElseWare creó algunas herramientas interesantes, por ejemplo FontSmart, FontWorks, Infinifont y PANOSE.

Infinifont es capaz de recrear un perfil a partir de los parámetros establecidos por PANOSE. Es un formato pensado para la sustitución de fuentes ausentes en el sistema, siempre que estén clasificadas con números PANOSE. El sistema es vectorial y modular, de ahí que las fuentes ocupen sólo un 10% del espacio en disco de una TrueType. El motor gráfico toma las especificaciones PANOSE, que le explican cómo es la letra, y coloca el módulo correspondiente sobre un esqueleto arquetípico.

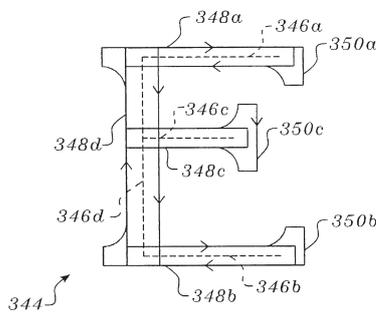


Fig. 7: Construcción de una E de caja alta en Infinifont; los números son claves PANOSE.

3.3. *Incubator*

Capaz de construir formas alternativas a partir de un perfil vectorial de entrada. Creado por Sampo Kaasila, creador de TrueType, basado en la tecnología de las instrucciones Delta, capaces de deformar los perfiles de las letras (perfil acomodado) para un sólo cuerpo y resolución.

Incubator toma un tipo PostScript o TrueType y le aplica diferentes cambios como si fueran instrucciones Delta, es decir, mueve los puntos de anclaje del perfil de la letra para generar una nueva figura. Los parámetros que el usuario podía cambiar son: peso, ancho, contraste, altura de x, descendentes y prosa. Existen pocas referencias sobre el producto, ya retirado de la circulación, pero sabemos que el programa tenía un modo estándar, con limitaciones para el usuario no instruido en tipografía, y otro avanzado, que requería cierto conocimiento del medio pues permitía la distorsión absoluta de los caracteres. Exact-Match era un parámetro capaz de aplicar la información contenida en el archivo .afm de un tipo PS Type 1 (grueso de cada carácter y pares de acoplamiento) a cualquier otra fuente del sistema.¹² Modular Words, una empresa de Man-

¹² De esta manera se podía crear un nuevo alfabeto que tuviera las misma prosa que una garamond o una bodoni, lo cual resultaba útil para sustituir fuentes no instaladas en el sistema o

chester, creó los tipos para la promoción del programa: TS Venetian y Teddy. Type Solutions, creada por el mismo Kaasila, es la empresa que comercializó el producto, entre 1992 y 1993, en sus diferentes versiones: Incubator, Incubator Pro. Más tarde, con el lanzamiento del formato TrueType GX apareció Incubator GX. Este último es muy interesante, pues es capaz de crear una fuente multi-eje en formato GX, similar a Multiple Master, pero sin tener que dibujar los extremos para la interpolación. Incubator trabaja por extrapolación. Bitstream compró la empresa el 2 de diciembre de 1998, y Sampo Kaasila pasó a formar parte de la directiva. Una vez más, un producto interesante, más incluso que el propio Chameleon, no sobrevivió a los peligros del mercado. Los fabricantes de software no podían permitir la existencia de un producto capaz de acabar con el negocio de venta de licencias tipográficas, pues los clientes sólo necesitarían la redonda y la cursiva para generar la familia completa, o generar una tipografía con la prosa de la Garamond sin tener que comprarla; las posibilidades que ofrecía a los diseñadores de letras no fueron tenidas en cuenta: 600,000\$ fue el precio que se le puso a la cabeza del enemigo.



Fig. 8: Capturas de pantalla de Incubator Pro 2.02. Cortesía de George Tomas

4. A PARTIR DEL 2000: LA DEMOCRATIZACIÓN DE LAS HERRAMIENTAS

El mundo ha cambiado mucho en estos últimos diez años, y la tipografía no es ajena a esta mutación: la generalización del uso de los ordenadores a escala planetaria, la web 2.0, el nacimiento de los *my media* (o los *self media*, según el autor) nos han convertido a todos en potenciales creadores de contenidos. Por otro lado empezamos a darnos cuenta de los nuevos usos que damos a la tecnología y cómo han cambiado las experiencias con las que disfrutamos.

Hoy por hoy dudo mucho que alguien se viera amenazado por programas como Chameleon o Incubator; el que desea una Bodoni y no quiere o no puede comprarla, la piratea directamente, no se pone a trabajar para emularla o reconstruirla. El concepto mismo del tiempo se ha vuelto tan rápido como la tecnología a nuestra disposición, desde los móviles a la informática, cuya potencia se

hacer pruebas en un documento sin variar la composición.

duplica cada 18 meses.

Esto implica la coexistencia de dos realidades bien opuestas: un mercado orientado a la calidad, a los tipos bien dibujados y programados, y otro enfocado hacia el divertimento fácil y rápido. Es en este último contexto en el que encaja perfectamente la tipografía orgánica, pues no implica un costoso aprendizaje y no requiere un uso prolongado para obtener resultados. De hecho, ni tan sólo se requiere instalar nada en la propia computadora, todas las operaciones se hacen en la nube de datos en la que se ha convertido internet.

Algunas de las iniciativas que han surgido en esta dirección corren a cargo de las mismas fundiciones digitales, quienes ya no ven este tipo de software como una amenaza sino como un complemento a su experiencia cultural. Por ejemplo, FontShop ha lanzado Fontstruct, un programa de tipografía modular que, en la misma línea del Súper-Veloz de Trochut, permite crear un tipo en pocos minutos a cambio de registrarse en su web. La estrategia es sencilla: conocer quién está interesado en tipografía a un coste ridículo, a cambio de dejar jugar un rato al usuario con el programa. En la misma línea, aunque requiere instalación en el disco local, DTL LetterModeller de la fundición Dutch Type Library, que funciona exactamente igual que el Ares Chameleon.

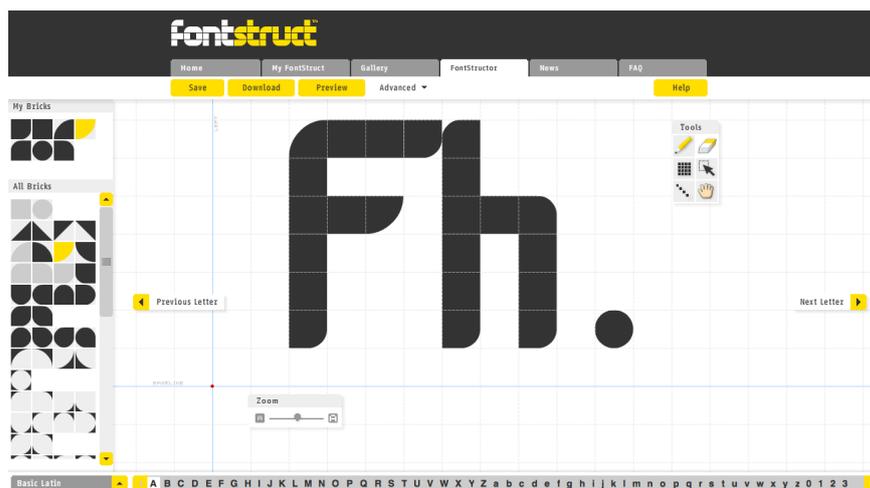


Fig. 9: Captura de pantalla de Fonstruct (FontShop)

Paralelamente hay una corriente denominada código abierto o software libre (GNU) en la que se inscriben cada vez más proyectos interesantes; es el caso de Kallculator (en proceso), Font Constructor o Constructor (Rodríguez 2007). El denominador común es acercarse al gran público, reducir el número de parámetros al máximo, aún a riesgo de empobrecer el resultado, para facilitar su uso a cualquier persona no instruida en tipografía. No se trata de producir tipos de la calidad de la Helvética, sino de adecuar las experiencias tipográficas al medio en el que se inscriben, al mismo tiempo que se abre el abanico de posibilidades permitiendo la entrada de conceptos como lo aleatorio, la interacción con otras entradas de datos como el sonido, el contexto, etc.

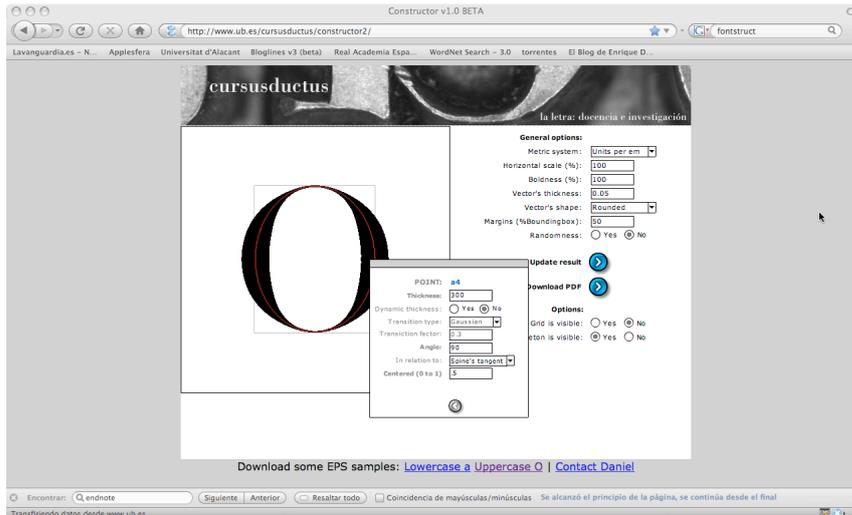


Fig. 10: Captura de pantalla de Constructor v 1.0 beta

Por último cabe destacar una extensión (*plugin*) para Adobe Illustrator llamado Live Pen, de Zero-One Inc. (año 2008), que permite construir formas caligráficas al estilo de Constructor y Kallculator. Lo que parece claro es el interés por lo caligráfico como sistema orgánico de escritura, que en estos momentos se traduce en la compra en 2009 por parte de FontLab —el programa de edición de tipografías profesional líder en el sector— de Live Pen, para incorporarlo en su sistema y convertirse en una herramienta híbrida, que permitirá por vez primera trabajar con tipografía sintética y orgánica sin salir de la aplicación.

5. A MODO DE CONCLUSIÓN

El exceso de complicaciones es uno de las causas principales que puede impedir que un programa obtenga difusión y alcance al mayor número de usuarios; hoy cualquiera es un diseñador en potencia, con lo que hay que repensar las herramientas para públicos diferentes a los de los años noventa. Las experiencias exitosas en nuestra sociedad mutante están cada vez más basadas en la rapidez, la superficialidad y la espectacularidad, los valores de las nuevas generaciones digitales.

Pero la informática es totalizadora, ofrece la posibilidad de trabajar desde dos puntos de vista tan distintos como lo orgánico y lo sintético. Ambas son perfectamente compatibles y deben coexistir en un mercado en el que hay sitio para todos. La tipografía sintética está más relacionada con la tipografía en plomo y por tanto mantiene los estándares de calidad industrial, detalle, laboriosidad e incluso su alto coste. La tipografía orgánica está emparentada con la caligrafía, es más fluida, viva, inmediata y directa. Está al alcance de todos, y puede llegar a ser tan fea o tan bella como habilidoso sea el que la construye.

Lo que parece claro es que es necesario evitar las valoraciones desde uno de los dos ámbitos, porque entonces habríamos de dejar al contrario fuera del campo, y necesitamos alguien que nos devuelva la pelota.

BIBLIOGRAFÍA:

- Fabian, Nicholas. *Ares: a model for typographic creativity* [en línea]. 29/04/2000. Disponible en <<http://web.archive.org/web/20000823070956/webcom.net/~nfhome/ares.htm>>. [consulta: 20 de mayo de 2005].
- Jacques y Hersch, Roger (ed.): *Actas del congreso Raster Imaging and Digital Typography*, Cambridge (Inglaterra), Cambridge University Press, 1989, p. 232
- Karow, Peter: «Automatic hinting for intelligent font scaling», en André, Kinch, Richard J. «Belleek: A Call for metafont Revival». *TUGboat*, 1998, vol. 19, núm. 3, pág. 247.
- Kindersley, David y Wiseman, Neil. «Computer Aided Letter Design». *Printing World*, Octubre de 1979.
- Knuth, Donald E. *Computer Modern Typefaces*. Reading (Massachusetts): Addison-Wesley, 1986. Pag. 1-7
- McQueen III, Clyde D. y Beausoleil, Raymond G. «Infinifont: a parametric font generation system». *Electronic Publishing*, Septiembre de 1993, vol. 6, núm. 3, pág. 117-132.
- Coueignoux, Philippe J.M.: *Generation of Roman Printed Fonts*. Ph.D. Thesis, junio de 1975. Reimpreso con permiso del Instituto de Tecnología de Massachusetts.
- Mergler, H. W. y Vargo, P.M. «Computer Assisted Letter Design». *Journal of Typographic Research*, Octubre de 1968, vol. 2, núm. 4.
- Rodríguez Valero, Daniel: «Propuesta de tesis doctoral sobre tipografía digital paramétrica o el diseñador gráfico como artesano del siglo XXI», *Primer Congreso de Tipografía*, (ponencia), Valencia 25-27 de junio de 2004. Edita la Asociación de Diseñadores Gráficos de la Comunidad Valenciana (ADCV), páginas 61-65.
- «Systèmes typographiques intelligents et souples», *Document numérique*, vol. 9, pp. 117-132, (2007)
-