Engineer

El subrayado que atravesaba todas las letras

28 de septiembre de 2026

Nuestro sitio subraya todos los enlaces y deja que el navegador abra un hueco donde el trazo descendente de una letra cruza la línea: la `g`, la `p`, la coma. Safari no lo hacía. La línea las atravesaba de lado a lado. Llevaba haciéndolo en las páginas en inglés y danés desde que el sitio era multilingüe.

La causa eran tres declaraciones de fuente para un alfabeto que esas páginas nunca descargan.

Aplique la ingeniería a su sistema

La ingeniería detrás de «El subrayado que atravesaba todas las letras» puede orientar su próximo cambio. Podemos investigar comportamientos similares en su software, datos o infraestructura. Interfaces que cargan rápido, funcionan en el teléfono y se comportan igual en los navegadores que utilizan sus clientes.

Explore nuestros servicios Póngase en contacto →

Siga el razonamiento y sus límites

Lea el razonamiento y sus límites en esta nota. Podemos comprobar si el mismo diagnóstico se sostiene en su entorno. Dirección del desarrollo integral de mapas GIS — cartografía integral, de la base de datos al navegador

Dirección del desarrollo integral de mapas GIS Póngase en contacto →

Lo que debería hacer el navegador

`text-decoration-skip-ink` es la propiedad que interrumpe el subrayado alrededor de un trazo descendente. Está activada de forma predeterminada y, desde hace años, la respuesta habitual ante un subrayado que se ve mal es recurrir a `text-underline-offset` y bajar la línea. Es el reflejo equivocado: moverla cambia el diseño; lo que faltaba era el hueco.

Encontramos dos causas reales antes de llegar a la interesante, y conviene conocer ambas.

- **Un subrayado de `1px` obtiene un hueco del tamaño de `1px`.** La interrupción que abre el motor es proporcional al grosor de la línea. Nuestra hoja de estilos fijaba ese grosor en un píxel, lo que dejaba unos 0,6 px de aire a cada lado de un asta: técnicamente un hueco, visualmente una línea que atravesaba la letra. Establecer `text-decoration-thickness: auto` escaló ambos a la vez y amplió el hueco de 3,5 px a 12,4 px sin mover el subrayado ni un solo píxel.
- `from-font` **no es una opción inicial tan segura como parece.** Lee el grosor declarado por la propia tipografía, y Ubuntu declara 0,056 em para Light, 0,020 em para Medium y 0,120 em para Bold: una familia cuya línea Medium tiene menos de la mitad del grosor de Light. A tamaños grandes, eso deja una línea de seis píxeles junto a otra de dos en la misma página.

Eso corrigió Chrome. Safari seguía trazando la línea por encima de todo.

La parte que más tiempo llevó

Lo medimos en condiciones: controlar un Safari real, cambiar el color del subrayado, capturar la pantalla a una escala ocho veces mayor y contar los píxeles donde la línea se detiene y vuelve a empezar.

	Grosor dibujado	**Hueco alrededor de un trazo descendente**
Chrome	2,00 px	5,8 / 12,5 / 6,3 px
Safari	1,38 px	1,8 / 1,0 / 1,0 px

Safari sí abría un hueco. Solo que era seis veces más estrecho que el de Chrome, y a tamaño de lectura un hueco de un píxel no se percibe como tal. Peor aún: el ajuste habitual lo empeoraba. Con un grosor de 3 px, Safari pasaba de tres huecos a uno; con 4 px, a ninguno, porque no escala el hueco con la línea. Lo mantiene en unos 0,07 em a cualquier tamaño, mientras Chrome concede 0,6 em.

Probamos unas veinte declaraciones: `skip-ink: all`, las dos formas antiguas de la propiedad, todos los grosores de medio píxel a cuatro, `text-underline-position`, `font-smoothing`, `text-rendering`, `paint-order`. Nada lo ensanchaba. En ese punto, la conclusión honesta era que WebKit limitaba el hueco y que no se podía hacer nada más desde una hoja de estilos.

La conclusión era errónea. La desmintió un informe de error en un proyecto de fuentes: Safari ignora skip-ink cuando una tipografía se carga con un subconjunto no latino.

La causa real

El sitio sirve Ubuntu en seis partes: tres grosores de caracteres latinos y tres de cirílicos. Así una página en ucraniano usa la misma tipografía que una en inglés, en vez de recurrir a la que ofrezca el sistema. Cada parte declara el rango de caracteres que cubre y el navegador descarga solo las que la página necesita.

Las seis se declaraban con un único nombre de familia, la forma evidente de escribirlo. Y ese es el desencadenante, registrado como error 255159 de WebKit: **cuando un estilo de una familia declara un rango de caracteres no latino, Safari degrada skip-ink para todos los caracteres de esa familia**, incluido el texto latino de una página que nunca solicita el otro archivo.

Una página en inglés mostraba mal los subrayados porque en algún lugar de la hoja de estilos existía una declaración de fuente para ucraniano. Borrar esas tres líneas en tiempo de ejecución, sin cambiar nada más, amplió los huecos de 1,8/1,0/1,0 px a 4,5/11,0/5,0 px.

La solución

Dé a la segunda escritura su propio nombre de familia e inclúyalo después del primero en la lista de fuentes:

```
@font-face {
    font-family: UbuntuCyrillic;   /* was: Ubuntu */
    src: url(ubuntu_300_cyrillic.woff2) format('woff2');
    unicode-range: U+0400-045F, U+0490-0491;
```

```
}

:root {
    --body-font: Ubuntu, UbuntuCyrillic, system-ui, sans-serif;
}
```

Una letra latina se encuentra en la primera familia y nunca llega a la segunda. Una cirílica pasa de largo por la primera, que no tiene ese glifo, y llega a la segunda en lugar de a una fuente del sistema. Los rangos de caracteres siguen decidiendo qué se descarga, así que la carga no cambia: las páginas en inglés y danés solicitan tres archivos latinos y ninguno cirílico; las ucranianas, lo contrario.

Tres declaraciones renombradas y una entrada en la lista de fuentes. Los huecos de Safari pasaron a 4,5/11,0/5,0 px; Chrome quedó intacto, el subrayado no se movió y el ucraniano sigue compuesto en Ubuntu con anchos idénticos.

Qué le diríamos a la siguiente persona

- **Una medición de skip-ink solo es válida para el motor que la produjo.** Teníamos una tabla de cifras, un diagnóstico correcto y una solución real, todo ello en Chromium. La captura que reabrió el caso vino de una persona que estaba mirando la página.
- **Pruebe el grosor antes que el desplazamiento.** Ensanchar el hueco conserva el diseño; bajar la línea lo sustituye.
- **Dividir una tipografía por escritura es correcto, pero la división también debe aparecer en el nombre de familia.** Compartir un nombre entre escrituras queda más limpio y cuesta skip-ink en Safari. Volver a unirlas parece una limpieza y es una regresión.
- **La carga nunca fue el problema.** La intuición dice que una página que se ve mal por una fuente cirílica debe estar descargando algo indebido. No era así. Bastaba la declaración.

Fuentes

- Error 255159 de WebKit: `text-decoration-skip-ink` y subconjuntos de fuentes
- Incidencia 1096 de Fontsource, que identificó el desencadenante
- MDN: `text-decoration-skip-ink`

- MDN: `unicode-range`

Todas las entradas del blog

Mida antes de cambiar

Describa el flujo de trabajo, las interfaces existentes y los usuarios que necesitan el cambio. Podemos definir una entrega comprobable y un traspaso que permita a su equipo mantener lo entregado.

Vea cuánto cuesta Póngase en contacto →

Hable de sus observaciones

Hablemos de «El subrayado que atravesaba todas las letras» en el contexto de su proyecto. Cuéntenos qué observó y cómo sería un resultado útil.

Explore nuestros servicios

welcome@engineer.company +45 50 10 61 84 Copenhague, Dinamarca

Llévese esta página

Tarjeta de contacto

Creado con esmero y cariño por D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Derechos de autor © 2025 – actualidad · Engineer ApS · Todos los derechos reservados.

Impreso desde https://engineer.company/es/notes/safari-underline-font-subset/