

The latest news and updates from the Google Drive team.

---

## What's different about the new Google Docs?

Tuesday, May 11, 2010

*Editor's Note: This post is more technical than most posts on the Google Docs blog.*

A month ago we [introduced](#) the latest version of the Google document editor. The new editor comes with features like a ruler, tabs stops, and floating images. Those features might seem pretty basic, but they're nearly impossible to support in a regular online text editor. This post unwraps some of the core technical changes with the new editor to make this new functionality possible.

### **The old Google documents**

As background, most online text editors (including the old Google documents) use an [editable HTML element](#), which means the application tells the browser to make a certain string of text editable, and the browser takes care of letting the user edit that text. So when you type in the old Google document editor, the browser inserts the characters you type into the page's HTML. Likewise, when you bold a word, the browser changes the HTML so that the word displays as bold.

Relying on the browser like this has several advantages:

1. Easy implementation -- Browsers know when a user triple clicks, they want to select an entire paragraph. The application doesn't need to think about these basic text behaviors.
2. Easy to make it fast -- The browser (not the app) handles the most computationally intensive task: text layout. Since layout is a core component of browser functionality, you can trust that layout performance has already been heavily optimized.

But using the browser's native text editing means less control over how the document behaves: if one browser has a bug in its list behavior, people using that browser will have trouble working with lists in Google Docs and we won't be able to fix the behavior for them. It also means we can support only the least common denominator of features: if inserting tabs works in some browsers but not others, we can't really support it because the doc won't look right if you open it in a browser that doesn't understand tabs.

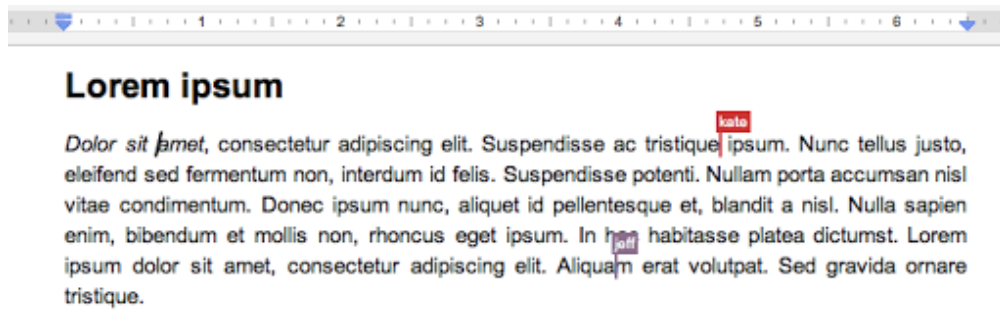
## The new Google documents

To get around these problems, the new Google document editor doesn't use the browser to handle editable text. We wrote a brand new editing surface and layout engine, entirely in [JavaScript](#).

### A new editing surface

Let's start by talking about the **editing surface**, which processes all user input and makes the application feel like a regular editor. To you, the new editor looks like a fairly normal text box. But from the browser's perspective, it's a webpage with JavaScript that responds to any user action by dynamically changing what to display on each line. For example, the cursor you see is actually a thin, 2 pixel-wide **div** element that we manually place on the screen. When you click somewhere, we find the x and y coordinates of your click and draw the cursor at that position. This lets us do basic things like slanting the

cursor for italicized text, and it also allows more powerful capabilities like showing multiple collaborators' cursors simultaneously, in the same document.

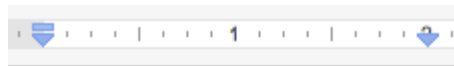


*Multiple users editing in the same paragraph*

## A new layout engine

By far the most difficult thing the editor does is figure out where to draw text. For this, we built a new **layout engine**. Here's an example of how the new engine works: say you type the letter 'a'. We notice you pressed the 'a' key and respond by drawing a single 'a' off-screen. We then measure the width and height of that 'a', combine those measurements with the x and y position of your cursor, and place the 'a' at the correct spot on the screen. If you're in the middle of a word, we push the characters after your cursor over. If you're at the end of a line, the editor moves your word to the next line and pushes any overflow to the lines after it.

Tab stops and other basic features are impossible to support if you're using the browser's HTML layout engine for your text. That's why we wrote our own engine: once we tell our layout engine how to draw a feature, we don't have to worry about which features browsers support.



## Burgers

All burgers are served with a  
drink and your choice of side.

Classic	\$8.00
Cheeseburger	\$9.00
<i>choice of cheddar, swiss, american</i>	
Bacon Cheeseburger	\$9.50
California Burger	\$10.50
Hickory	\$9.00
<i>(pickles, onions, hickory sauce)</i>	
Veggie Burger	\$8.50

*The formatting in this basic menu couldn't be supported without writing a new layout engine*

### Improved collaboration

What I've just described is pretty standard architecture for a desktop word processor. But the new Google Docs isn't just an online version of existing desktop software: it's designed specifically for character-by-character real time collaboration. That kind of collaboration is only possible because we built the editor around a technology called [operational transformation](#). It's what lets multiple people edit the same area of a document at the same time without needing to wait for the server to say a particular edit is okay.

Building an extensible, fully collaborative online word processor required rewriting every part of the document editor from scratch. We're still adding more features and polish before turning it on for everyone, but for an early peek, you can opt-in by visiting the **Editing** tab in the [Google Docs settings](#).

Posted by: Jeff Harris, Product Manager, Google Docs



Google

[Google](#) · [Privacy](#) · [Terms](#)