Fat themes: a problem with WordPress paid themes

A common problem with WordPress paid themes is that they're often bloated. This causes several performance issues, both in the frontend and in the backend. There's no quick solution to this problem, but there are several approaches you can adopt to mitigate the side effects of using a theme which requires too many resources.

You can see what I mean with the word "bloated" simply by activating your new theme on a website that previously used the default WordPress theme. Pages take a while to be fully loaded, JavaScript actions and events seem to lag, images all of a sudden slow down the document's rendering and so on.

What's happened? To answer this question, let's take a look at how these themes are developed.

First, fat themes often use a framework that is hooked into the WordPress flow. The problem with frameworks is that they force WordPress to load their entire structure every time you access a section of your site. It doesn't matter if you're in the admin section or not: the framework is always there.

Some of these frameworks use an OO approach, others don't. From a pure performance perspective, results are the same: if there are too many dependencies, your site will slow down. If you have a root access to your site, you can try to increase the resource limits used by PHP. Alternatively, you have to manually modify your theme.

Second, things get even worse on the client-side. Fat themes typically load all the assets (CSS and JavaScript files) even when it's not necessary.

For example, if you have a Google Map on your contact page, these themes load the remote API's script on all the sections of your site. This means that a GET request to Google's servers is performed also on the home page.

To mitigate this problem, you can use a CDN or the Apache's caching system. This approach doesn't fix the problem, though. In fact, assets will be always loaded in the same wrong way.

You get what you paid for, right? In this case you actually get too much.