



## Hooks Explained

### What is it?

A hook system is provided by Loaded Commerce to allow your add-on to 'hook into' the rest of Loaded Commerce; that is, to call functions in your add-on at specific times, and thereby set your add-on in motion.

### How it Works

Loaded Commerce Add-ons hook into the core using a virtual file modification system known as vQmod. The concept is quite simple... Instead of making changes to the core files directly, the changes are created as xml search/replace script files. These script files are parsed during page load as each "source" core file is loaded with the "include" or "require" php functions. The source is then patched with the script file changes, and saved to a temp file. That temp file is then substituted for the original during execution. The original source file is never altered. This results in a "virtual" change to the core during execution without any actual modification to the core files.

### Features

- No actual code changes are made. All changes are "virtual".
- Modifications are stored in their own files and applied "on-the-fly" at runtime.
- Instant Single file "plug-n-play". Add the file to apply the mod, remove the file to remove the mod.
- No worries about losing custom core changes during upgrades.
- Multiple modifications can be made to the same file without conflict.
- Easily update or enhance customizations without having to edit any code.
- Full visual of the actual changes taking place in the generated temp files for debugging.
- Fails gracefully back to the original source file if there is an error.
- Exceptional logging option to track every change made.
- Simple structured xml format. (See readme for full breakdown of xml syntax)
- Multiple options for find/replace, regex, positions, offsets, indexing, error handling, and more!

### Performance

The way the code is designed has reduced any effect on performance. In our page parse time tests, we have seen absolutely no performance change when enabled or disabled. When the actual source files are parsed, only the files that have changes need to parse the xml scripts. All in all, performance surprisingly does not appear to be a factor.