**Client /Server side Target**

Server-side tools render code on the server-level and send a randomized version of the page to the viewer with no modification on the visitor’s browser. Client-side tools send the same page, but JavaScript on the client’s browser manipulate the appearance on both the original and the variation.

Client-side tools use either synchronous or asynchronous code, both of which have pros and cons regarding page speed, flicker effect, and loading issues.

Client-side tools are easier for marketers to use, and you can often get tests up and running faster without burning development resources. However, server-side tools offer greater flexibility and control. They also protect your privacy and security better, and for web apps, reduce the need for ‘heavy pages.’

Both tools have their use cases and don’t need to be used exclusively. However, far more important than the tool you choose is the process by which you optimize.

**Which type is your tool?**

| VER-SIDSserver side | client sideCLIENT-SIDE |
| --- | --- |
| [Conductrics](https://conductrics.com/) (both capabilities) | [AB Tasty](https://www.abtasty.com/) (both capabilities) |
| [Adobe](https://www.adobe.com/marketing/target.html) | [Conductrics](http://conductrics.com/) (both capabilities) |
| [SiteSpect](https://www.sitespect.com/) (both capabilities) | [Convert](https://www.convert.com/) |
| [Optimizely](https://www.optimizely.com/) (both capabilities) | [Bound](https://www.bound360.com/) |
| [AB Tasty](https://www.abtasty.com/) (both capabilities) | [Kameleoon](https://www.kameleoon.com/en) |
|  | [Monetate](https://monetate.com/) |
|  | [Omniconvert](https://www.omniconvert.com/) |
|  | [Optimizely](https://www.optimizely.com/) (both capabilities) |
|  | [Oracle Maxymiser](https://www.oracle.com/marketingcloud/products/testing-and-optimization/) |
|  | [VWO](https://vwo.com/) |

**Client Side Advantages:**

* Client-side tools offer ease of use, faster execution, and a WYSIWYG editor. You don’t necessarily need to take up your development team’s time to get [experiments](https://cxl.com/blog/learning-analyzing-experiments/) running.
* The advantage of testing on the client side is speed and simplicity. You can test a lot of changes quickly without much initial investment. On the other hand, testing on the server side is both more work and generally more powerful.
* Client-side tools let marketers run tests accurately without the help of developers.
* “When thinking about speed, there are two different factors that matter. The first is after your initial deployment and set-up, how fast can you get a test from concept to execution to live? The goal for any program is to get most tests with at least five experiences through that concept in 30 minutes or less. That is a hard target to meet but it does express just how important it is to prioritize speed in testing and the need for general knowledge of CSS, HTML, and JavaScript by the operator.

**Server Side Advantages:**

* Run experiments for websites with dynamic content.
* Test non-UI changes that still have an affect on your objective. For example, a database query result set that is returned to a user.
* Integrate [Google Analytics](https://cxl.com/blog/google-analytics-102/) experiments with your service (e.g., content management provider).
* Manage experiments using your own optimization platform.

Debugging:



