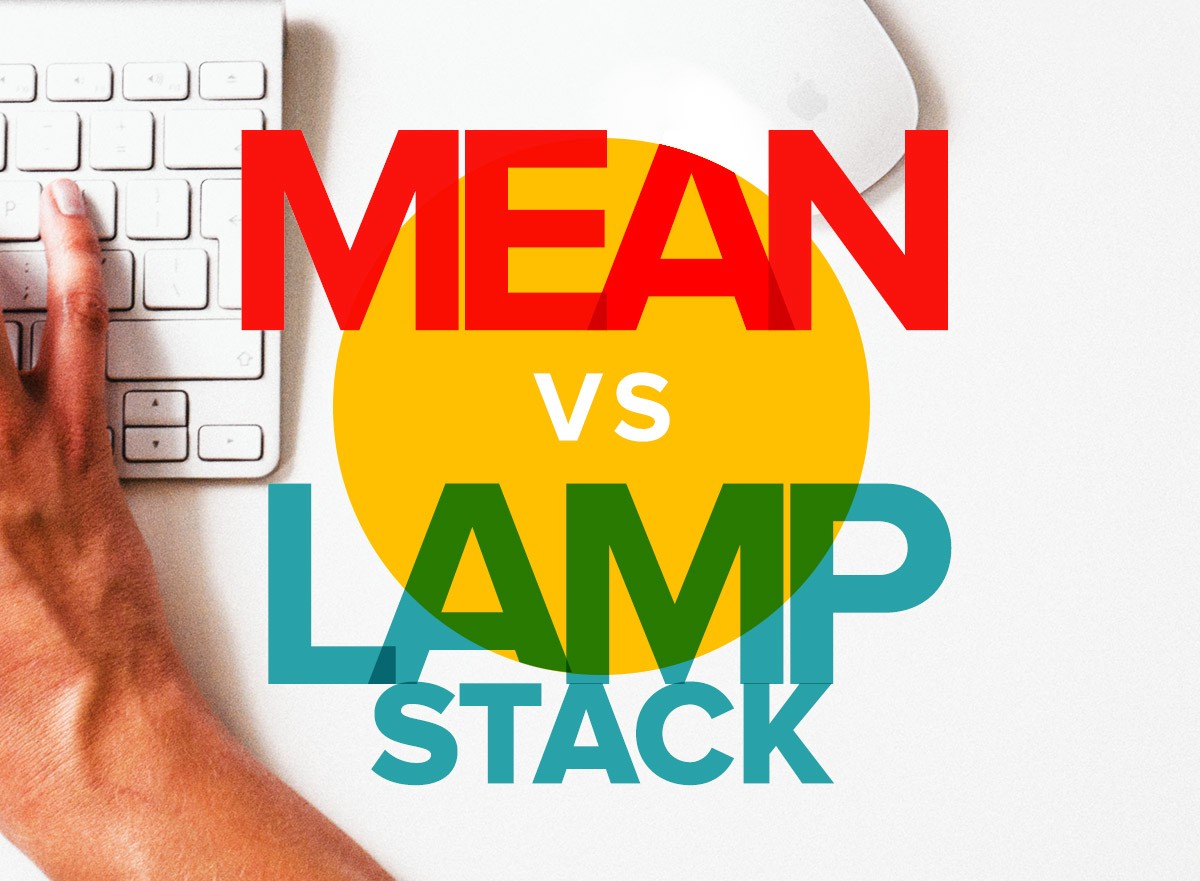
**MEAN vs LAMP: Choosing the Right Stack for Your Startup Project**



The MEAN vs LAMP argument has been going on for some time now, and MEAN seems to be winning the popularity contest. But does this mean that it’s the end of LAMP? Far from it because when it comes to free open source web development stacks, [LAMP still held in high regard](http://www.logicrux.com/lamp-solution-stack/) in web development circles.

Choosing the right stack for your startup will come down to the developers that you have on board or offshore and your specific business needs.

**What’s MEAN?**

MEAN is a software bundle that stands for MongoDB, ExpressJS, AngularJS, and NodeJS. Together in a stack, these free programs enhance the simplicity of the web development process.

**What’s LAMP?**

LAMP, on the other hand, stands for Linux, Apache, MySQL, and PHP, Perl, or Python. It’s still relevant as it offers a great alternative to commercial software packages.

Further, it works well as a bundle of programs that provide a robust platform to develop and implement web-based applications and servers. For years, it has been the most effective solution to develop [enterprise level web apps](http://intersog.com/portfolio/web-apps/) with enhanced customization and flexibility, cost-effectively.

So what’s with all the confusion? Let’s take a closer look.

**A Closer Look at MEAN**

MEAN basically describes an application stack:

MongoDB (data storage)

Express.js (server-side application framework)

AngularJS (client-side application framework)

Node.js (server-side language environment although Express implies Node.js)

MongoDB provides a lot more [flexibility to store data](http://opensourceforu.com/2016/04/mongodb-a-document-oriented-database/). Further, Express.js helps to standardize the web building process. AngularJS offers a clean and simple way to add interactive functions, while Node.js affords an improved nexus for running your server.

Stacked together they offer a clean and coherent mechanism to move data back and forth from user to disk farm.

**A Closer Look at LAMP**

The open source nature of LAMP interactions between end users and the source can be potentially modified to suit the developers’ specific needs. Further, MySQL and PHP make it easy to code and even [novices can easily build something](https://www.codementor.io/learn-programming/how-to-build-app-from-scratch-beginner-programmer) and get it up and running quickly.

As it uses PHP as a standard APACHE module and uploads PHP files through a MySQL Database to an APACHE server, deployment is a breeze. However, LAMP is essentially an acronym for a classic web server stack:

Linux (OS)

Apache (web server)

MySQL (data storage)

PHP or Perl or Python (scripting language)

Comparatively, MEAN is an ideal solution for JavaScript lovers as it’s a powerful stack with diverse capabilities. Further, the database layer is also completely replaced with JSON (native data language of JavaScript), while storage is achieved via MongoDB.

**So What’s the Best Option For Your Startup?**

Again, the answer to the question lies in the specific needs of the business. Both stacks have their own set of advantages and disadvantages, so the one you choose will largely depend on the type of web application that you’re looking to build.

A key advantage of choosing MEAN is the fact that JavaScript code can be reused for both the server and the client. But this is rarely ever seen in practice because it will help maintain security if the client and server code bases are kept separate.

But using JavaScript for both the front-end and the back-end makes life easier for [developing teams](http://intersog.com/services/software-development-team/) to switch between the two. It makes the workflow homogeneous and also enables two groups of full stack developers to work well together on the application as a whole.

It’s no secret that it can be quite a challenge to switch between PHP or Python for the server and then use JavaScript and HTML for the client. So if you’re more comfortable with JavaScript than other languages, MEAN is the right choice for you.

But on the other hand, end users often use ad blockers and disable JavaScript because of privacy concerns. Most browser extensions also allow the users to actively select the scripts that are permitted to run, so your app won’t work if it’s completely based on JavaScript.

Disabling JavaScript has rendered many websites useless, so the battle continues between users and websites. This has even led to some users being banned from certain websites as this type of user behavior can break web applications.

So although JavaScript comes with the advantage of making your web app look modern, it might not be a feasible solution to reach all users. Further, JavaScript also slows down websites, so its popularity can diminish significantly in the near future. So all this has to be taken into account when making your stack selection.

However, you can also avoid going heavy with JavaScript at the front end and take advantage of the speed offered by Node.JS. But at the same time, PHP is also just as fast when optimized.

Traditional multi-threaded Apache setups are resource heavy, so LAMP can also be an option that ends up being slow. But PHP can be significantly optimized by Nginx or Lightspeed. Yet, replacing Apache with Nginx can be quite a headache to configure.

Choosing MEAN over LAMP will provide your development team with the benefit of enhanced speed for data retrieval, flexibility in deployment, and a single language that’s used from top to bottom. But it also comes with its own set of disadvantages.

Both software development stacks are going to continue to play an important role in the development of web applications for years to come. But the ultimate choice for your startup will come down to the use-cases and business priorities for the app that’s under development.