

1. **Jenkins**

- a. Able to scale to support a large number of nodes
- b. Jenkins supports all OS and versions
- c. Comes as a WAR file that just needs to be dropped into a JEE container, ready for quick setup
- d. It comes with a web interface so setting up is simple and easy
- e. Disturbing work is easily spread across many machines

Getting Started

- i. Jenkins link: <https://www.jenkins.io>
- ii. Jenkins has a for documentation and download on their main page
 - 1. Jenkins Doc link: <https://www.jenkins.io/doc/>
 - 2. Easy to use documentation page with easy access to get started and gain understanding of all aspects of the product
 - 3. Link to using Jenkins: <https://www.jenkins.io/doc/book/using/>
 - a. Extremely easy to get started and using the product with the easy steps and documentation on the site.
 - b. There is a large number of resources for each step of understanding the product and getting the product installed and ready to go
 - c. Jenkins Sandbox can be used for many different types of testing/experimentation
 - 4. Downloading Jenkins link: <https://www.jenkins.io/download/>
 - a. This will take you to the download page for the product
 - b. It will give two options with slightly different environments

2. **Big Eval**

- a. Embedable into automated dataops and devops projects
- b. High quality gates for system component deployment and data streams in and out of data stores
- c. Autopilot testing for agile development of analytical data models, datamarts, and data warehouse
- d. High performance in-memory scripting and rules engine
- e. Abstraction for any kind of data
 - i. Big Eval link: <https://go.bigeval.com>
 - 1. Two links on the site home page to start free trial and use sandbox. Also, has a link to speak to a live expert
 - a. Home page educates to the shire size of the products capabilities
 - 2. Getting started link: <https://bigeval.com/platform/testdrive-options/sandbox/>
 - a. After filling out your information, the free demo and sandbox is available. You get a number of resources to understand the product start up and download to more advanced features.

<u>extraLargeArray</u>	
Insert	1.06 s
Append	4.1 ms
<u>tinyArray</u>	
Insert	35.7 us
Append	94 us
<u>smallArray</u>	
Insert	105.7 us
Append	201 us
<u>mediumArray</u>	
Insert	170.9 us
Append	132.8 us
<u>largeArray</u>	
Insert	8.8 ms
Append	565.9 us

Results

Append pre-test

Without running the test I can see that append will be faster due to using the .push method as it will add to the end .push (stacking to the array without moving other locations) instead of the beginning (moving all parts of the array down one) .unshift.

Append post-test

After running the test it is shown that the .push (append) method is faster with larger arrays but smaller arrays are faster with the .unshift (insert) method.