The purpose of the research is to establish the effectiveness and efficiency of Node.js as a development tool/environment not only in terms of its now well documented performance, an example of which is published by Lie et al. in their 2014 paper for the International Conference on Computational Science and Engineering (Lie, et al., 2014), but also in terms of the ease of development, including the full range of development tools and frameworks that are available in other languages, including scaffolding, database connection/communication and de-bugging. Notwithstanding Ionnas K. et al’s conclusion that end-to-end JavaScript is a viable option for building modern web apps (Chaniotis, et al., 2014), Netflix, another high profile internet goliath, discovered that their Node based application was eating up CPU resources while developing hourly endpoint latency increases of 10ms (Xiao, 2014), indicating that it's not all plain sailing for applications running on Node.js and leads me to ask the question:-

"***What factors should a developer consider when deciding to build a web application in Node.js and at what point are the obvious performance benefits of Node outweighed by the practicalities, impracticalities and potentially costly and unforeseen pitfalls involved with the implementation of this, still relatively new, technology?***"

I feel text in bold could form the basis of my Question???

The plan is to develop a simple application such as a messaging/chat app and/or an image processing app in three different language platforms; Node.js, Ruby on Rails and PHP. The common denominator amongst these three technologies is that the server side scripting is interpreted.

There have been many examples in industry of Node.js based applications outperforming other, more traditional technologies as detailed by PayPal in the November 2013 PayPal Engineering Blog post Node.js at PayPal server side where almost unbelievable stats were achieved offering massive performance benefits from an application built in less time with fewer people, with less code and fewer files by comparison with an equivalent application being built in Java with more people, more code, more files and offering substantially less in terms of performance (Harrell, 2013).

When comparing different technologies it’s important to define specific criteria which are firstly relevant to the domain in which these technologies are to be used, equally important is that the criteria be measurable in some meaningful way so that the ensuing results can then be used by others to inform decisions and choices they might be making in relation to the technologies in question. One example of a set of comparison criteria can be found in Matt Raible’s 2010 presentation titled Comparing JVM Web Frameworks (Raible, 2010)

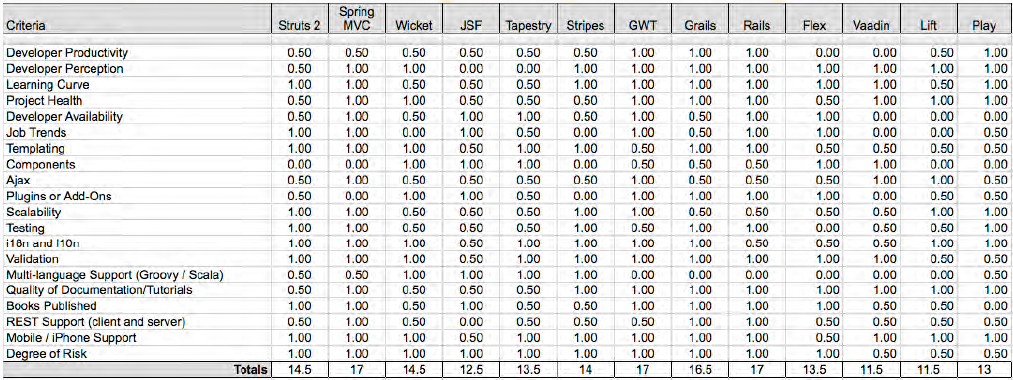


Figure 1: Example of Comparison Points (Raible, 2010)

Siddharth lists 15 points to consider when choosing a Development Framework (Siddharth, 2009), amongst which are:

* Usage Context
  + Which states that the chosen framework should be suitable and appropriate in terms of meeting the requirements of the application being developed.
* Software Pattern
  + Whilst MVC is the predominant pattern to be found behind most commercial apps and web apps, certain use cases in your application may demand something more specific or bespoke.
* Hosting
  + This may have most relevance to a developer who is planning for certain reasons to develop an app using Node.js, as Node will only run on a server using the Google V8 engine (Nodejs, 2015). Whilst the software is free to download,

As is outlined

Mongrel???

Rails has zero concurrency

Merb

# Bibliography

Chaniotis, I. K., Kyriakou, K.-I. D. & Tselikas, N. D., 2014. Is Node.js a viable option for building modern web applications? A performance evaluation study. *Computing,* 2014(March), pp. 1-22.

Harrell, J., 2013. *Node.js at PayPal.* [Online]   
Available at: http://www.paypal-engineering.com/2013/11/22/node-js-at-paypal/  
[Accessed 14th August 2015].

Joyent, 2014. *Error Handling in Nodejs.* [Online]   
Available at: https://www.joyent.com/blog/best-practices-for-error-handling-in-node-js  
[Accessed 15th August 2015].

Lie, K., Ma, Y. & Tan, Z., 2014. *Performance Comparison and Evaluation of Web Development Technologies in PHP, Python and Node.js.* s.l., IEEE.

Nodejs, 2015. *Node.js.* [Online]   
Available at: https://nodejs.org/  
[Accessed 19th August 2015].

Raible, M., 2010. *Comparing JVM Web Frameworks.* s.l.:Raible Designs.

Siddharth, 2009. *15 Most Important Considerations when Choosing a Web Development Framework.* [Online]   
Available at: http://code.tutsplus.com/tutorials/15-most-important-considerations-when-choosing-a-web-development-framework--net-8035  
[Accessed 19th August 2015].

Thamsen, L. et al., 2012. *Orca: A Single-language Web Framework for Collaborative Development.* s.l., International Conference on Creating, Connecting and Collaborating through Computing.

Thomas, D. & Hansson, D. H., 2005. *Agile Web Development with Rails.* 2005-7-5 ed. s.l.:The Pragmatic Programmers.

Xiao, Y., 2014. *The Netflix Tech Blog.* [Online]   
Available at: http://techblog.netflix.com/2014/11/nodejs-in-flames.html  
[Accessed 14th August 2015].