

# Like Cars on a Highway.



Don MacVittie, 2012-31-01

Every once in a while, as the number of people following me grows (thank you, each and every one), I like to revisit something that is fundamental to the high-tech industry but is often overlooked or not given the attention it deserves. This is one of those times, and the many-faceted nature of any application infrastructure is the topic. While much has changed since I last touched on this topic, much has not, leaving us in an odd inflection point. When referring to movies that involve a lot of CGI, my oldest son called it “the valley of expectations”, that point where you know what you’d like to see and you’re so very close to it, but the current offerings fall flat. He specifically said that the Final Fantasy movie was just such a production. The movie came so close to realism that it was disappointing because you could still tell the characters were all animations. I thought it was insightful, but still enjoyed the movie.



It is common to use the “weakest link in the chain” analogy whenever we discuss hardware, because you have parts sold by several vendors that include parts manufactured by several more vendors, making the entire infrastructure start to sound like the “weakest link” problem. Whether you’re discussing individual servers and their performance bottlenecks (which vary from year to year, depending upon what was most recently improved upon), or network infrastructures, which vary with a wide variety of factors including that server and its bottlenecks.

I think a better analogy is a busy freeway. My reasoning is simple, you have to worry about the manufacture and operation of each vehicle (device) on the road, the road (wire) itself, interchanges, road conditions, and toll booths. There is a lot going on in your infrastructure, and “weakest link in the chain” is not a detailed enough comparison.

In fact, if you’re of a mathematical bent, then the performance of your overall architecture could be summarized by the following equation:

$$\text{Performance Degradation} = \sum_{i=1}^n (\text{performance degradation (Infrastructure Element } i \text{ )})$$

Where n is the number of infrastructure elements required for the application to function correctly and deliver information to the end user. From databases to Internet connections to client bandwidth, it’s all jumbled up in there. Even this equation isn’t perfect, simply because some performance degradation is so bad that it drags down the entire system, and other issues are not obvious until the worst offender is fixed. This is the case in the iterative improvement of servers... Today the memory is the bottleneck, once it is fixed, then the next bottleneck is disk, once it is improved, the next bottleneck is network I/O... on and on it goes, and with each iteration we get faster overall servers.

And interestingly enough, security is very much the same equation, with the caveat that a subset of infrastructure elements is likely to be looked at for security, just because not everything is exposed to the outside world – for example, the database only need be considered if you allow users to enter data into forms that will power a DB query directly.

$$\text{Security Degradation} = \sum_{i=1}^n (\text{security degradation (Infrastructure Element } i \text{ )})$$

So what is my point? well simply put, when you are budgeting, items that impact more than one element – from a security or performance perspective – or more than one application, should be prioritized over things that are specific to one element or one application. The goal of improving the overall architecture should trump the needs of individual pieces or applications, because IT – indeed, the business – is built upon the overall application delivery architecture, not just a single application. Even though one application may indeed be more relevant to the business (I can’t imagine that eBay has any application more important than their web presence, for example, since it *is* their revenue generation tool), overall improvements will help that application *and* your other applications.

Of course you should fix those terribly glaring issues with either of these topics that are slowing the entire system down or compromising overall security, but you should also consider solutions that will net you more than a single-item fix.

Yes, I think an advanced [ADC](#) product like F5's [BIG-IP](#) is one of these multi-solution products, but it goes well beyond [F5](#) into areas like SSDs for database caches and such.

So keep it in mind. Sometimes the solution to making application X faster or more secure is to make the entire infrastructure faster or more secure. And if you look at it right, availability fits into this space too. Pretty easily in fact.

---

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | [f5.com](#)

F5 Networks, Inc.  
Corporate Headquarters  
[info@f5.com](mailto:info@f5.com)

F5 Networks  
Asia-Pacific  
[apacinfo@f5.com](mailto:apacinfo@f5.com)

F5 Networks Ltd.  
Europe/Middle-East/Africa  
[emeainfo@f5.com](mailto:emeainfo@f5.com)

F5 Networks  
Japan K.K.  
[f5j-info@f5.com](mailto:f5j-info@f5.com)

---

©2016 F5 Networks, Inc. All rights reserved. F5, F5 Networks, and the F5 logo are trademarks of F5 Networks, Inc. in the U.S. and in certain other countries. Other F5 trademarks are identified at [f5.com](#). Any other products, services, or company names referenced herein may be trademarks of their respective owners with no endorsement or affiliation, express or implied, claimed by F5. CS04-00015 0113