Building a new breed of ADC

F5's Paul Szabo describes the birth of VIPRION, the first on-demand Application Delivery Controller.

Paul Szabo, a product development architect with F5, is a part of the team that created VIPRION, an Application Delivery Controller (ADC) that uses modular performance blades to scale and add power as needed. Szabo spoke with *F5World* Editorial Director Paul Desmond about the story behind VIPRION and how his team addressed some vexing computer science problems to create it.

Paul Szabo (front) and the VIPRION VIP team.


The VIPRION team in Spokane, Wash., pose with the world's first on-demand Application Delivery Controller.

Q What is the history behind the development of VIPRION?

A It dates back to 2000 with our former CTO, Carl Amdahl. Processors were doubling their speed every 18 months, and chip sizes kept getting smaller. We knew eventually we would want to add multiple CPUs to each system, but our software architecture at the time was limited mostly to running on one CPU. To scale up to heavy-duty processing, meaning 10 Gbps of throughput, we needed to be able to scale in parallel, like a supercomputer. The project has been brewing ever since then, under the code-name Montreal, and has been through two or three hardware iterations. Finally, three years ago, we decided we had the right software infrastructure in place, so we got to work on what became VIPRION.

Q Where do you see VIPRION fitting into the market?

A An obvious target for this is at the high end of the market. Monster web sites and service providers are seeing traffic double every six to nine months. You can stick more ADCs in there, but then you've got a management headache, because you've got to manage all those boxes. With VIPRION, you just pop a new blade in and you're done. And you're still managing one box. There are also enterprises out there, however, that may not need the performance of four blades today. If they have VIPRION with one or two blades, they now have a growth strategy that will carry them several years into the future.

Q VIPRION blades are also hot-swappable, with automatic failover. What kind of technical challenges did that present?

A There's a concept called ACID [Atomicity, Consistency, Isolation, Durability] that allows you to know for sure whether you've received an entire configuration change. Much of our control and management planes weren't ACID. If they're not ACID, and in the middle of a transaction I pull out a blade, then the system becomes inconsistent and unstable. So we basically spent two years fixing all the parts that weren't ACID.

Q VIPRION certainly acts like a supercomputer, with up to 16 CPU cores forming a pool of processing power that can be used on an as-needed basis. What was involved in pulling off that feat?

A There's a law in computer science called Amdahl's Law [Ed. note: The law is named for Gene Amdahl, Carl's father]. Every time I process a packet, the more interactions I have with other CPUs processing other packets, which means I can't scale. The idea is to make the CPUs not interact at all, giving you true, almost linear scalability. That provides almost perfect localization, and there's no overhead involved in adding a second, third, or fourth CPU. It's also making sure the various buses, such as PCI buses, are balanced, so they don't get oversubscribed.

Q How do you know when a product is finished? There must be a temptation to keep adding features and functions.

A For VIPRION, we tried really hard not to add features beyond those we already had in the BIG-IP system. We wanted the product to be just like managing a BIG-IP. If we added a bunch of features or changed a lot of GUI screens, we would have defeated that purpose.

Q So it will be simple for BIG-IP users to make a leap to VIPRION?

A Exactly. That's been the goal from Day 1. ✳