riverbed
Think fast.

# RIVERBED ON SOFTWARE DEFINED NETWORKING

**What does it mean to say that a network is "software defined"? How does such a network behave differently from a traditional network—if at all? Routers and switches run software, Riverbed® hardware appliances run software—so is "software-defined networking" (SDN) nothing more than a marketing exercise?**

At Riverbed Technology, we see SDN neither as pure hype nor as the destination of interest. SDN presents some intriguing opportunities and is an intermediate step to a more-significant destination: the software-defined data center (SDDC). Indeed, the SDDC is the real prize, creating opportunities for organizations to achieve new forms of flexibility that have so far eluded traditional data center practices. While SDN has some interesting technical characteristics, it is properly understood as a prerequisite and component of SDDC—a means to an end, rather than an end in itself.

In this paper, we position SDN as one of the crucial elements for meaningful network virtualization and a core component of the larger trend toward programmable infrastructure. Additionally, we present a high-level sketch of how Riverbed intends to move toward delivering the complete vision of the software-defined data center. To realize that vision, we plan to deliver products with increased portability, greater scale density, richer programming interfaces, novel approaches to licensing, visibility into SDN protocols, and an understanding of SDN performance feedback loops.

## THE JOURNEY SO FAR: INFRASTRUCTURE BECOMES CODE

Software on "ordinary" computers has been replacing dedicated hardware for some time now. The relentless march of Moore's Law has allowed general-purpose computers to supplant a growing number of specialized networking devices. At the same time, the definition of application has expanded beyond its traditional data processing roots and now includes the functions formerly reserved for those displaced devices. These developments bring greater flexibility, new forms of programmability, and better resource efficiency into modern data centers.

- **Greater flexibility.** The general-purpose computers supporting "infrastructure applications" can offer greater flexibility when compared to dedicated devices. Functionality can be added, upgraded, enhanced, or removed without replacing the hardware. A notable characteristic is a marked increase in function decomposition and service abstraction. What was once a monolithic construct is now decomposed into distinct functional units, each of which performs a specific operation. As an example, consider some of the developments that have occurred in public cloud computing. Originally, "rental" of virtual machines by the hour was the predominant choice; customers still had to provision several virtual machines on which to install all the necessary resources for applications. Now, providers offer database tables, queuing/notification/message services, block and object storage, search facilities, and more. The same underlying hardware supports all these capabilities.
- **New levels of programmability.** Deployed as software on general-purpose computers, network infrastructure components often expose application programming interfaces (APIs). Through a scripting language or other automation tool, developers can incorporate network functions directly into applications. Continuing with the earlier cloud example, basic load balancing—once available only through expensive hardware appliances—is a service offered by nearly every provider. Some providers offer firewall, prioritization, and VPN services. APIs allow developers to provision, manage, and decommission such services on demand.
- **Better resource efficiency.** What once required space in a rack along with associated power and cooling can now be accomplished with a few lines of code. As a result, a given amount of hardware can enable improved efficiency and support higher concentrations of workloads. Physical constraints show no signs of abating soon: space is scarce and energy remains costly. Significant cost savings come from consolidating physical resources while increasing the quantity of available upper-layer services. And, most importantly, developers gain a degree of placement optimization not previously possible. Services can be placed in the most appropriate location at the most appropriate time.

As more and more services transition from hardware into software, a natural next step is to transition the entire data center into software. For that to happen, the network itself must be decomposed into abstracted services.

## THE NEXT DESTINATION: VIRTUALIZING THE NETWORK

Virtualization of memory, disk, and CPU long ago moved into the mainstream. While early styles of virtualizing a network, such as VLANs and simple virtual switches, are commonly used, traditional network infrastructure components have mostly retained their everything-in-a-box characteristics. Depending on your point of view, decoupling network functions from the hardware has been seen either as threat or simply not necessary. Yet as enterprises rush to emulate clouds, the importance of such decoupling becomes evident. Portable workloads are built entirely from logical abstractions, and true portability is possible only when the "network" comes along for the ride.

### WHAT IS VIRTUALIZATION?

**Broadly, it's the creation of a conceptual version of a real thing. In the context of IT, virtualization is a method of decoupling software and services from the underlying hardware and repackaging resources as abstractions that exhibit the same characteristics as hardware.**

Consider a virtual machine. Essentially, it's a pile of software that exhibits the characteristics of a physical machine. Can a virtual network be similarly constructed, as a pile of software? Perhaps more fundamentally, is that a useful thing to do? Before answering these questions directly, let's recall the benefits that arose from the move to virtual machines. It isn't just about multiplexing several workloads across general-purpose machines. More importantly, a virtual machine offers an abstraction that's just right for the data center: provisioning, moving, snapshotting, and rolling back are thorny in hardware, but easy in software. A "software-defined computer" changes the operational model and streamlines management of workloads.

Clearly, virtualizing servers is useful. Is it useful to virtualize networks? In a sense, this has already happened, but with limitations. VLANs can virtualize network segments, but can't synchronize state counters or access control lists (ACLs) when a virtual machine moves from one segment to another. A more useful virtual network would encapsulate all the required operations into a single abstraction. This should include not only segments and tags but also counters, ACLs, policies, and so on. The logical abstraction for a virtual network enables the same kinds of operational conveniences we see for virtual machines.

Given that virtual networks are useful, is it possible to construct a network from software? Conceivably yes, but it turns out that this has traditionally been a difficult thing to accomplish. A software-based network should:

- Decouple the virtual network from the physical network
- Be viewed by upper-layer services as identical to a physical network
- Permit meaningful units of operation (just like server virtualization)

Recently, changes in network equipment design promise to convert this vision into reality. To understand virtual networks, start by thinking about the two major conceptual components of a traditional router or switch: a forwarding plane that moves traffic from one interface to another, and control plane that learns network topology and provides instructions to the forwarding plane. In traditional physical network equipment, the scope of a switch's control plane is bounded by the box that contains it—an arrangement that's inflexible, may have topology mandates or constraints, and can't easily adapt to change or recombination.

In virtual networks, the control plane is decoupled from the physical container. The forwarding plane continues to exist in hardware, but no longer makes decisions—it simply does what it's told to do. The control plane is repositioned into a distinct and abstracted service, often implemented as software running on general-purpose computers. This software replaces the traditional control plane, and doesn't rely on the usual routing protocols to instruct the forwarding plane how to move traffic around the network. The control plane becomes centralized and possesses a full end-to-end view of the entire network. This improved visibility provides more information for making forwarding decisions than is possible with the hop-by-hop nature of traditional routing protocols.

While these ideas are not exactly new, SDN presents some novel approaches to network virtualization. By changing the focus from open protocols to open APIs, SDN-based virtual networks enable new degrees of programmable flexibility. For example, SDN exposes opportunities to create network controllers that are not constrained by any particular choice of wire protocol. No longer limited to purchasing network equipment from a small number of incumbent vendors, network designers will be able to choose from a variety of third-party controllers, seeking those that best match an organization's business and technical requirements. SDN extends the virtual network beyond routing and switching into upper-layer services such as traffic control, data protection, and even applications. Network controllers can make decisions from a much wider source of metrics than traditionally available.

Building a virtual network without SDN is certainly possible, but probably not as useful. Virtualization maps multiple logical networks across a common physical fabric. As changes occur in the logical and physical worlds, the abstractions represented by logical networks must remain consistent. Sophisticated state management becomes a challenging technical problem when logical networks could be located just about anywhere. It turns out that SDN is very good at managing large numbers of states while providing reasonable degrees of consistency. It maintains connection and flow state independent of the hardware, and easily tolerates changes to the forwarding plane. Without the state management afforded by SDN, the operational utility of network virtualization diminishes considerably.

## SDDC BECOMES THE NEW NORMAL FOR ENTERPRISE NETWORKS

"Full stack" virtualization characterizes the software-defined data center. Network, compute, and storage resources are made available as abstract services to applications, and are decoupled from the underlying hosting hardware. SDDCs blur the distinction between on-premise and cloud; indeed, SDDCs possess all the necessary attributes that can enable true cross-cloud workload portability.

Increasingly, proper application deployment requires finely tuned infrastructure to support it. Such tuning is becoming more and more application specific, including targeted QoS policies, just-in-time resource allocations (to cope with demand spikes), transaction awareness (for cost accounting purposes), and differentiated network paths. Monolithic network equipment cannot accommodate these diverse application-specific requirements. SDDCs eliminate large infrastructure boxes and replace them with network services dedicated to and tuned to the needs of individual applications.

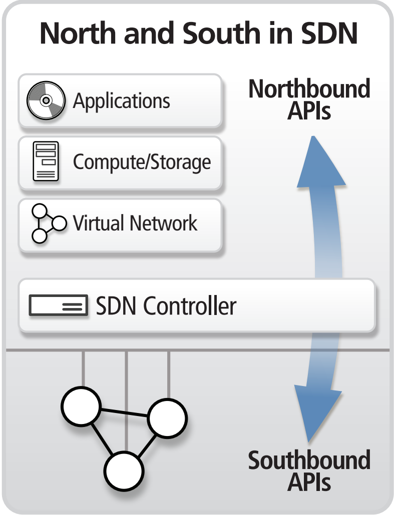


**Figure 1. Relationships between SDN controller APIs.**

In SDDCs, developers take center stage. Emerging APIs enable developers to create recipes that define the business requirements, infrastructure requirements, service levels, and cost considerations for modern workloads. Orchestration layers above ("north of") the controller marshal virtualized network, compute, and storage resources as appropriate; developers need not concern themselves with the technical details of how this happens below ("south of") the controller (see Figure 1). This illustrates the true power of abstractions and decoupling: changes can be made at any particular layer without affecting the operation of adjacent or distant layers.

SDDC transcends the plumbing to encompass the whole of IT, including applications. SDDC changes the limits of where IT can happen. Applications can be decomposed into compute workloads, transport workloads, and storage workloads; these various tasks can be executed wherever makes the most sense. Time-sensitive workloads can be placed into an environment with very fast (but likely more expensive) processors and storage, while cost-sensitive workloads can be placed into an environment that reduces speed and expense. Developers can indicate requirements in their code, and the programmable nature of an SDDC ensures that those requirements are met by automatically assigning resources on demand. New possibilities for efficiency, security, and availability arise as a result.

riverbed
Think fast.

# RIVERBED ENABLES AND ORCHESTRATES THE SDDC

Our first goal is to help make the transition to SDDCs as seamless as possible. Our second goal is to ensure customers realize as wide a range of efficiency, security, and availability benefits as possible. With these in mind, here are of some of our thoughts as we move in the direction of SDDCs:

**RIVERBED AT A GLANCE**

STEELHEAD
*WAN optimization*

GRANITE
*Storage delivery*

STINGRAY
*Application delivery*

WHITEWATER
*Cloud storage*

RPM
*Performance management*

**Broad programmability.** Today, all Riverbed products provide web-based and command line configuration. For programmable configuration, we are developing a management plane API called FlyScript™ that will operate across the entire product line. Currently available for the Stingray™ family, and Cascade® Shark and Cascade® Profiler appliances, FlyScript presents interfaces to perform provisioning, configuration, maintenance, and reporting. (We are already seeing interesting use cases targeting specific verticals; often, our customers share them for reuse.) Work is underway to expand to Steelhead®, Granite™, and Whitewater® products. Data and control plane programmability present additional opportunities for automation. Available now is Stingray TrafficScript®, a powerful scripting language for on-the-fly traffic inspection and modification that's also easy to learn. Similar programming capabilities for other products are currently under evaluation.

**Increased portability.** We will continue to make our products more portable and adaptable. The SDDC environment can be incredibly diverse, and we will run wherever the SDDC exists. This could be in a private cloud, a public cloud, a disaster recovery site, or some combination of these. In all such scenarios, the computing and storage platforms are not homogeneous. To provide maximum coverage, our products will become agnostic with respect to operating system, hypervisor, and cloud provider. Today, the Stingray family (encompassing application delivery and control, web security, and web content optimization) is available on all modern hypervisors and can run as a guest on both Linux and Windows hosts. Virtual Steelhead® appliance (for WAN optimization) is available for ESX and Hyper-V. Virtual editions of Whitewater (for cloud storage) and Granite™ Core (for storage delivery) round out the current offerings. This trend will continue across the entire product portfolio.

**Density as a scale metric.** Traditional IT environments reward "speeds and feeds." Space is scarce and power is expensive in modern data centers, and not all workloads require maximum speeds and feeds. A more important metric in SDDCs is density: maximizing the number of instances of a given service on any particular compute or storage node. A feature of Stingray Traffic Manager called "micro ADC" allows customers to deploy a much greater number of application delivery controller instances per compute node than ever before, bringing on-demand per-application traffic management to reality—while preserving the isolation expected in multi-tenant environments. Steelhead® Cloud Accelerator software follows a similar concept for high-density, per-tenant WAN optimization. As the Riverbed product portfolio evolves, high-density instance deployment will become common.

**License aggregation.** Nothing about SDDC can be static, including the licenses that govern software deployments. Licenses for products in the Stingray family are portable, meaning they can be moved to different servers when capacity changes are required. We will innovate across the entire Riverbed portfolio with new licensing models that reflect the agility of SDDCs. This will take the form of aggregate licenses that can be distributed across multiple instances regardless of where and how they reside. Enterprises need the ability to right-size their WAN optimization or ADC needs and elastically reallocate resources or alter capacities as workloads change, and we will respond to this need across product lines.

**SDN and virtual network visibility.** Typical SDN-based networks overlay hundreds or even thousands of on-demand tunnels between hypervisors. Traffic on the wire will look radically different than on a traditional network, and gaining visibility requires new tools. We will broaden the SDN support available in our application performance management (APM) and network performance management (NPM) products. Cascade products already understand the traffic within VXLAN networks, meaning that VXLAN-based networks can benefit from the added visibility our solution brings. We will extend this capability to include other north of controller protocols. Furthermore, we will add SDN support to the remainder of the APM product suite, extending visibility into the hypervisor to help customers troubleshoot and diagnose hypervisor-based root causes.

**SDN and SDDC closed-loop feedback.** An intelligent control plane is a critical foundation of a fully virtual network. Useful intelligence fed into the control plane permits efficient, timely, and cost-effective traffic distribution, which in turn improves application availability and response time. The best source for gathering such information comes from continuous measurements of the current state of applications, compute and storage demands, and network flow behaviors. Adding application and network visibility into SDNs and SDDCs eliminates the guesswork sometimes associated with compute and storage placement, transforming the control plane into a true network-wide director. FlyScript will gain capabilities to take action based on visibility—in essence, to act as a programming language for the management and control planes. FlyScript will integrate with and enhance existing control and orchestration frameworks such as OpenFlow, OpenStack, and CloudStack.

# RIVERBED ENHANCES THE PERFORMANCE OF AND VISIBILITY INTO THE VIRTUAL NETWORK

When an application stack is decomposed into distinct workloads running on disparate compute elements and storage resources, it may no longer be appropriate to assume that all resources are local. Multiple simultaneous locations will become the norm, thus WAN optimization will be needed to maintain a LAN-like "network dial tone" and to project a storage endpoint from one place to another. As applications running across disparate networks may be subjected to varying degrees of availability, throughput, and security, application delivery controllers (ADCs) developed for high-density multi-tenancy will become more relevant. Monitoring the growing number of applications that move about many disparate locations will become crucial, especially when such monitoring can be used to apply on-demand policy changes and to automate many of the manual processes that exist in today's IT infrastructure. The increase in the overall density and complexity of the traffic mix will reinforce the need to control bandwidth and protect business critical workflows. The diversity of locations combined with the surge in data movements will accelerate the adoption of hybrid WAN topologies.

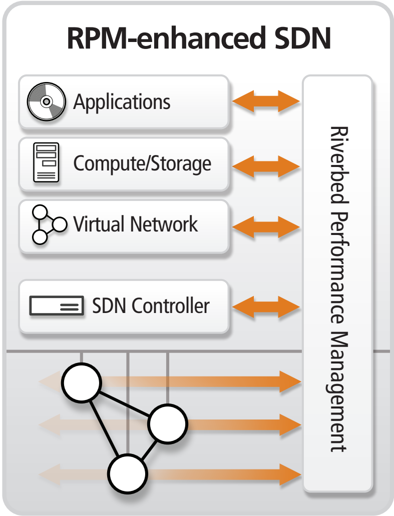


**Figure 2. RPM enhances SDN with greater intelligence.**

Our WAN optimization, application delivery, and storage delivery product families become network applications that easily insert themselves into the design of an application or service. They will operate largely unchanged and continue to provide measurable value. WAN optimization can, for instance, become a feature that a developer may choose to enable whenever large amounts of data must be moved rapidly or when user response time must be similar to that of a LAN. Developers can use SDDC orchestration tools to provision Riverbed solutions into a chosen design. These solutions will be fully virtualized, and can accommodate relocation whenever necessary. The addition of WAN optimization and application delivery help ensure a consistent user experience and high availability regardless of location.

We intend for FlyScript and the Riverbed Performance Management (RPM) portfolio of products to become a complete and accessible platform for developing, deploying, and maintaining network applications in SDDCs and atop SDNs. Recall the characterization of the SDDC as a collection of layered abstractions. At the SDN layer, network controllers are distant from business logic, and implementing custom network applications in the controller creates undesired dependencies. Instead, business logic should remain decoupled to preserve reusability. FlyScript facilitates the translation of upper-layer business logic into lower-layer constructs that the SDN then uses to manage network plumbing. Our RPM portfolio will provide much-needed intelligence that surrounds the controller, continuously collecting, consolidating, and reporting information about the current state of the full SDDC stack (see Figure 2). Applications and services at all layers can act upon this information to make automatic adjustments to policies and configurations.

As we help shape the future of infrastructure for distributed applications, we will build on the SDN technology stack to realize our vision. Consider, for example, a cloud-based management system for executing global bandwidth arbitrage, performing branch office path selection, evaluating and enforcing information security policies, and measuring application performance behaviors. This structure parallels the separation of control and data planes proposed by SDN. In fact, a cloud-based management system can integrate well with the control plane of an SDN and its APIs, both north and south of the controller.

## CONCLUSION

Whenever significant change occurs in network design, opportunities arise to evaluate and incorporate performance-enhancing technologies. The move to software-defined data centers represents such an opportunity. The operation of our WAN optimization and application delivery products does not change in such an environment and our value remains strong. The programmable nature of our solutions allows them to become useful and important elements in fully-virtualized network architectures.

Network virtualization also presents new challenges with respect to visibility and control, and our performance management solutions are well positioned to manage the traffic within and between virtualized networks. By helping to answer questions like "What's inside that overlay?", "Where should the infrastructure place this collection of virtual resources?", "How much bandwidth should be allocated?", and "What path is best for these flows?" we will enable faster and smoother adoption of SDDCs.

SDN and SDDC remain at early stages in their development. To fully realize the vision presented in this paper, emerging standards must solidify. However, we do not intend to remain idle waiting; instead, we will participate when it makes sense for us and our customers. At the same time, we will innovate by offering SDDC-enabling solutions that continue to deliver the performance customers expect as they realize their own SDN and SDDC visions.

## ABOUT RIVERBED

Riverbed delivers performance for the globally connected enterprise. With Riverbed, enterprises can successfully and intelligently implement strategic initiatives such as virtualization, consolidation, cloud computing, and disaster recovery without fear of compromising performance. By giving enterprises the platform they need to understand, optimize and consolidate their IT, Riverbed helps enterprises to build a fast, fluid and dynamic IT architecture that aligns with the business needs of the organization. Additional information about Riverbed (NASDAQ: RVBD) is available at **www.riverbed.com**.

**Riverbed Technology, Inc.**
199 Fremont Street
San Francisco, CA 94105
Tel: (415) 247-8800
**www.riverbed.com**

**Riverbed Technology Ltd.**
One Thames Valley
Wokingham Road, Level 2
Bracknell. RG42 1NG
United Kingdom
Tel: +44 1344 31 7100

**Riverbed Technology Pte. Ltd.**
391A Orchard Road #22-06/10
Ngee Ann City Tower A
Singapore 238873
Tel: +65 6508-7400

**Riverbed Technology K.K.**
Shiba-Koen Plaza Building 9F
3-6-9, Shiba, Minato-ku
Tokyo, Japan 105-0014
Tel: +81 3 5419 1990