Written by Lori MacVittie | Technical Marketing Manager, Application Services

Aligning Application Infrastructure with Business through Service-Oriented Application Delivery

Overview A successful SOA implementation relies as much on the flexibility of your network as it does the services which will be delivered on it. Just as brittle, inflexible application integration technologies have been deemed unsuitable and unable to meet today's volatile business needs, so too have the static, inflexible network technologies of the past become a hindrance to the delivery of dynamic, flexible service-oriented applications.

Challenge Without the ability to understand the unique needs of the services and applications that comprise your SOA implementation, the network can become the bottleneck that prevents you from realizing the full potential of your investment. An intelligent, application-aware network is a necessity to ensure that the benefits of your SOA implementation are not lost due to performance, availability, and security related issues.

Solution A service-oriented application delivery controller provides the flexibility required to support the dynamic nature of today's business and IT environments. By providing the means through which delivery policies can be applied to both services and applications, a service-oriented application delivery controller enables IT to meet its obligation to the business and supports business goals of agility, reuse, and risk mitigation. The service-oriented application delivery controller is itself service-oriented—centralizing shared application, network, and security services in a high-performance network device capable of supporting a wide variety of delivery scenarios. It therefore understands the unique challenges encountered when implementing a service-oriented architecture.

A service-oriented application delivery controller ensures that the network infrastructure supporting your SOA implementation aligns with the business, just like the services and applications it delivers. The right service-oriented application delivery infrastructure enables the SOA and the business to realize the benefits of agility, reuse, and risk mitigation.

Agility

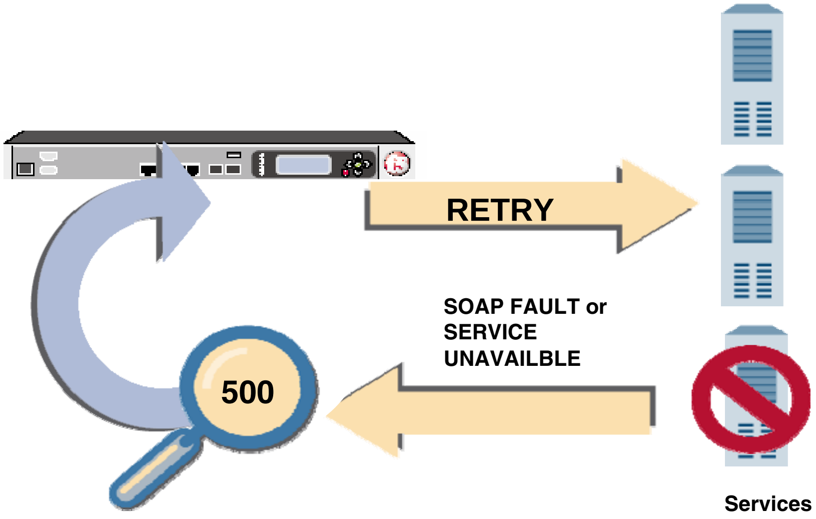
Agility of the business depends on the ability of IT to rapidly change business processes and business logic to adapt to volatile market and business conditions. IT enables that agility through the use of meta-data driven SOA-focused applications comprised of services that can be rapidly developed, tested, and deployed within an agile infrastructure framework.

Agility, in a simpler view, is (1) accepting that change will invariably be required, and (2) putting into a place an infrastructure that can both support rapid change as well as adapt to changing business and IT needs itself.

Modern development methodologies like Agile, XP, and Scrum do not address at what point in the development lifecycle architectural designs should be made, but all agree that change is inevitable. From an agility standpoint, a complete architecture should not be analyzed and developed at the beginning of the project. A loosely defined architecture capable of adapting to changing needs throughout the lifecycle of the project is necessary.

A service-oriented application delivery controller is flexible and meets the business' demand for agility by providing the means through which the delivery of applications and services can be adapted to meet delivery challenges without requiring the patches, updates, or upgrades of other network devices—all of which can interrupt service delivery and disrupt business.

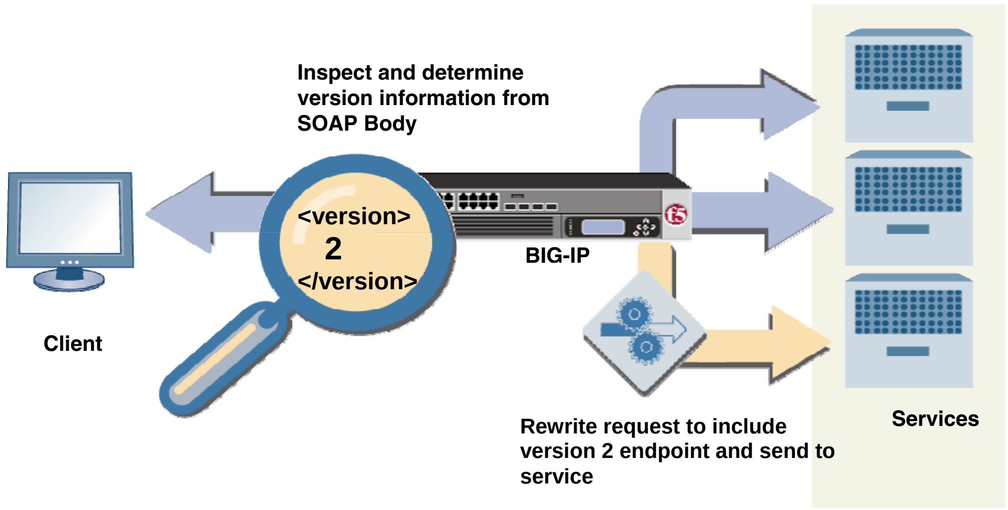
F5 addresses the issue of change in several ways. First and perhaps foremost, it provides the basic infrastructure through which service-oriented applications are assured delivery and availability. F5's BIG-IP Local Traffic Manager (LTM) ensures availability through advanced load balancing capabilities and extends those capabilities further with its unique iRules capabilities. Because BIG-IP LTM is a full proxy, it is capable of bi-directionally inspecting and manipulating content, and can further act upon that content in myriad ways. For example, if a service is unavailable or returning a SOAP Fault that indicates a problem with the service, an architect can use an iRule to send the request to another instance of the service and report the issue by alerting an administrator. This allows applications relying upon the service to continue to be available while IT addresses the root cause of the service failure.



By utilizing application-aware health-checks to validate that a service is working properly as well assessing its state of availability, F5's BIG-IP LTM can also automatically divert requests to those services capable of responding at the time of the request. LTM is capable of making such decisions in a dynamic environment based upon real-time conditions such as service response time or current load on the service. The ability to react dynamically to changing conditions in the service ecosystem ensures the availability of individual services and, by extension, applications.

In addition to its core competency in minimizing the impact of change on service availability, F5's iRules and iControl APIs can assist in addressing the fluid nature of an agile IT architecture. The flexibility iRules provides IT in deploying the underlying infrastructure necessary to support an SOA allows rapid changes and unanticipated architectural issues to be addressed in an agile manner. Consider an environment in which change is introduced in version two of a service that requires modification of its interface or in existing message schemas. In many instances it is not feasible to enforce a cut-over date and force clients to move from version one to version two. A more flexible quiescent-based approach is often desirable. But in the interim it becomes necessary to support two versions of the same service, which may require architectural changes in both the network and applications.

F5 addresses this issue by providing the means by which application architects can implement a solution in a network-hosted device. Using iRules, architects can easily implement content-based routing to direct requests to the appropriate version of the service. This removes the requirement for clients to specify which version of the service they wish to use – this is determined automatically based on the content of the request they are sending. This means no external changes are required, as the endpoint for the service remains the same regardless of the version invoked, reducing confusion for service consumers. Using a statistics profile, administrators can track the usage of each service in order to determine when usage of the earlier version has dropped to a business determined acceptable level so that support of that version can be dropped.


Inspect and determine version information from SOAP Body
<version>
2
</version>
Client
BIG-IP
Rewrite request to include version 2 endpoint and send to service
Services

iControl offers unprecedented interaction between the infrastructure and applications deployed on that infrastructure, giving applications input and even control over the way in which it is delivered. iControl's standards-based interface (WSDL) offers access to authorized applications and the ability to perform a wide variety of application delivery tasks. Applications and administrators can offer access to these tasks based on administrative domains, or roles. This allows architects, developers, and administrators the ability to configure and deploy applicable policies and processes in the application delivery controller without affecting the core delivery network. Tasks such as provisioning new services, adding application-specific processing through iRules, modifying the status of services for maintenance or upgrade processes, as well as myriad other tasks can all be accomplished remotely through iControl.

The ability to dynamically modify the way in which applications are delivered allows the delivery network to support rapid change and accommodate the dynamic nature of SOA implementations.

Reuse

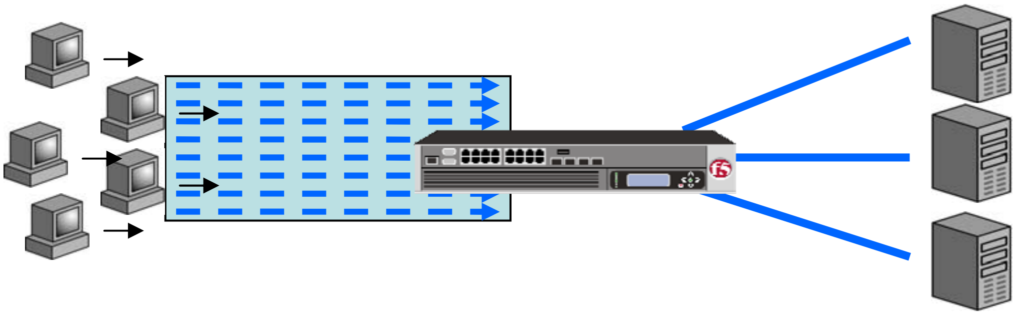
The goal of reuse is to cut costs, reduce cycle times, and ensure a level of consistency in business processes through reuse. Analysts estimate an average of 20% savings in development costs through the reuse of existing services.

Moving common business logic into a common set of reusable services, often called shared services, increases the burden placed on servers providing those services as they are, as their name implies, shared across a number of applications. Where the computational cost of the business logic was once distributed across servers, it is now located on a single server, but still being used by the same number of applications.

The additional burden on the server providing the shared service can potentially incur further costs through failure to meet defined service level agreements or general availability issues. Just as mission-critical applications require an architecture supporting high-availability, so too do mission-critical services. A solution that ensures availability and enables services to meet defined SLAs is paramount to achieving the full benefits of reuse and shortening the time in which the ROI of the shared service is fully realized.

Reuse of services also requires that certain application services be abstracted from within the service to reside elsewhere in the infrastructure, such as security. Because a single service may be shared by multiple applications, all with unique security policies, an external provider capable of applying the appropriate security policy at the appropriate time is necessary. This is also true for application-specific service level agreements, as these policies may differ from application to application even though they make use of shared services.

An application delivery controller meets these challenges by employing a variety of optimization techniques at the TCP and HTTP level. By allowing a BIG-IP application delivery controller to manage the myriad network connections required between shared services, it can relieve the burden imposed by TCP connection management on the server through technology such as its OneConnect connection pooling, improving capacity of the server by up to 30%. In many cases this effectively removes the network impact of shared services on the server.



BIG-IP application delivery controllers monitor and manage services and applications through a variety of standard-features and can assist in ensuring services are meeting defined service level agreements. Because BIG-IP devices are application-aware, they understand individual application needs and can adjust traffic routing decisions based on the unique needs of each application and service. Coupled with a network-aware application communicating via BIG-IP's standards-based iControl API, applications can adjust traffic management and routing decisions based on application and business specific parameters, providing additional control over the delivery of your service-oriented applications and shared services.

Because BIG-IP is an application delivery platform on which shared services can be deployed, network-focused services such as caching, authentication, acceleration, and transport layer security can easily be deployed. In addition to the benefit of reuse offered by this deployment option, it has the added benefit of providing a higher level of operational efficiency; many of these functions are optimized on application delivery controller but not on application servers.

It is further possible to utilize an application delivery controller to provide a platform on which shared application-specific services can be deployed. Common tasks such as data-validation and scrubbing can easily provide reusable, consistent execution of this application logic in the network. BIG-IP application delivery controllers accomplish this through iRules, which allows developers, architects, or administrators to deploy this common, shared logic. This has the added benefit of improving performance of application servers as this redundant task is offloaded to the network. This reuse supports the SOA goal of agility, as a change to this logic needs only be implemented in a single place, reducing the time needed to deploy the change as well as the possibility of introducing errors.

Mitigating Risk

One of the factors driving business to adopt SOA is the ability to mitigate risk through tighter controls over business processes. Several regulations, all inherently arbitrary and subject to change over time, impose serious financial penalties upon organization's failing to properly adhere to their directives. Sarbanes-Oxley (SOX), the PATRIOT Act, and Basel II mandates change that drive IT implementations.

Many companies lack the visibility into their business operations required by these regulations and turn to SOA as a mechanism through which granularity and visibility can be achieved.

Through the ability to control business processes, establish and enforce enterprise-wide security, privacy, and implementation policies, and ultimately provide auditable information trails are examples of how SOA can be used to mitigate risk within the enterprise.

SOA achieves these goals by breaking applications and business processes into their composite services, which can then be individually managed from a security perspective. Access control, audit trails, and privacy policies can be implemented in a modular and more granular way—one that does not involve massive changes to applications. SOA is inherently policy based, with policies defining access, encryption and decryption requirements, and even whether or not a full capture of messages should be enabled. This type of external policy enforcement is agile and allows the organization to adapt to changing conditions in regulatory requirements. It decreases the impact on applications by removing the enforcement of policy from inside the code to an external enforcement point, and makes deployment of such policies a more manageable process.

An application delivery controller capable of implementing and enforcing applicable pieces of these overarching policies can provide a more manageable process as well by centralizing the application of policy and controlling access to services. As with reuse, an application delivery controller can assist in mitigating risk while improving overall performance of services by offloading a number of functions from the application endpoints. Secure transmission of SOA messages is a requirement in many vertical industries, but the performance degradation associated with the most common mechanism for securing transport, SSL, is always painful. Similarly, the costs associated with managing and maintaining the certificates required for SSL has a negative impact on already limited budgets.

An application delivery controller centralizes certificate management, thus reducing the overall cost of certificates and the ongoing associated costs of maintenance, renewal, and management, as well as offloading the compute costs associated with processing SSL. Additionally, an application delivery controller such as F5's BIG-IP devices can assist in enforcing policy at the edge of the network, thus preventing the need to duplicate code to enforce those policies in every service. Data scrubbing, which implements policy requiring due diligence by organizations to prevent sensitive data such as Social Security numbers or credit card numbers from being delivered to clients, can easily be implemented at the edge of the delivery network and applied consistently across all services, thus ensuring compliance with state and federal regulations.

While this functionality can certainly be implemented in each service needing to comply with regulations, this is an inefficient method of enforcing policy and results in a reduction in agility should regulations change or require additional security measures. This aligns application delivery controllers with both business and IT needs, and is a benefit to both in terms of compliance, management, and overall improvement in performance due to centralization and offload of this redundant but necessary task.

Conclusion

One of SOA's primary goals is to align IT with the business. As the service delivery network through which applications are delivered is a part of IT, it makes sense that these platforms should also support those goals. Application delivery controllers such as F5's BIG-IP devices can and do assist in aligning IT with the business by enabling agility, mitigating risk, and encouraging reuse. As part of the IT infrastructure, these devices can improve overall performance, enhance security, and offload tasks from both servers and developers, providing quantifiable benefits over alternative solutions.

About F5

F5 Networks is the global leader in Application Delivery Networking. F5 provides solutions that make applications secure, fast and available for everyone, helping organizations get the most out of their investment. By adding intelligence and manageability into the network to offload applications, F5 optimizes applications and allows them to work faster and consume fewer resources. F5's extensible architecture intelligently integrates application optimization, protects the application and the network, and delivers application reliability—all on one universal platform. Over 10,000 organizations and service providers worldwide trust F5 to keep their applications running. The company is headquartered in Seattle, Washington with offices worldwide. For more information, go to www.f5.com.