NetApp White Paper

Infrastructure for Web 2.0/LAMP Applications Powered by NetApp and MySQL

Ravi Animi, NetApp
Gopinath Sankaran, MySQL
May 2008 | WP-7041-0508

ABSTRACT

The transition from static business-to-consumer (B2C) Web sites to dynamic B2C Web sites with rich content—essentially the transformation from Web 1.0 to Web 2.0—is enabling increased interactivity between businesses and consumers. Web 2.0 can be thought of as the technologies and Web sites that enable users and developers to collaborate in rapidly developing data and applications with a high level of integration across platforms and other services. Businesses that can capitalize on this opportunity to provide a richer and more useful experience can increase customer retention and loyalty to their brand. However, if a company is not able to quickly, flexibly, reliably, and securely deliver rich interactive content, the benefits afforded by this new level of interactivity go unrealized.

At the core of all LAMP (Linux®, Apache, MySQL, and PHP/Perl/Python) deployments is the application data repository (MySQL) and storage (typically NetApp® IP storage). Many IT organizations rely on data management services to help ensure the reliability, availability, and scalability of their static Web-based applications. Unfortunately, traditional networked storage solutions have not evolved fast enough to meet the delivery challenges posed by today's sophisticated interactive content.

This white paper discusses current B2C Web 2.0 trends and explains the data management challenges associated with dynamic content from a database and storage perspective. It introduces the requirements for the storage and delivery of dynamic content, so that IT organizations can extract the full potential of their investment in these advanced, feature-rich Web 2.0 applications. And it profiles two solution providers at the core of the Web 2.0 application infrastructure, NetApp and MySQL.

TABLE OF CONTENTS

1 EVOLUTION FROM WEB 1.0 TO 2.0

As enterprises increasingly depend on their Web sites to enhance top-line revenues and retain brand loyalty, they are discovering that complex site functionality and fresh site content are imperative to maintaining consumer interest and loyalty. Fortunately, today's Web site and infrastructure technologies support these objectives.

A quick review of the evolution of content delivery to distributed users explains why today's Web sites offer a higher level of interactivity between businesses and consumers. In the 1980s, companies relied on the client/server application computing model to deliver rich content to users. A fat-client GUI played a central role in the consumption of this content. This model required considerable investment in administration and infrastructure overhead and maintenance.

With the advent of the Internet in the 1990s, enterprises could deliver content to a much broader audience without time-consuming administration overhead and high maintenance costs. Although the Web browser interface essentially served as a thin client, the ability to deliver rich content was still limited by Web standards, Web browser functionality, networking and storage infrastructure technologies, and end-user access speeds. These older-generation Web application technologies generated static content that rarely changed.

Yahoo! is an excellent case study in the evolution of these technologies and the content they enabled. The original Yahoo! home page, launched in 1994, was little more than a directory listing of hyperlinks. (See Figure 1 for a brief survey of the evolution of Web technologies since 1990.) Web visitors clicked the links and read the page content—that was the extent of their online experience. Today's Web sites, a mix of static and dynamic content, allow visitors to interact with site content, as evidenced by Yahoo!'s newest design—a testament to information organization, multiple information formats, and interactive persona. It emphasizes rich user experience and personalization. In addition to browsing everything from news to video, users can access localized and personalized services such as Yahoo! Messenger, Yahoo! Mail, Yahoo! Maps, and Yahoo! Movies.

RICH INTERNET APPLICATIONS

Rich Internet applications (RIAs) are written using technology such as Adobe Flash, Adobe Flex, and AJAX (Asynchronous JavaScript™ and XML). RIAs enable rich content to be delivered to a broader audience of consumers. RIAs have the interactive interfaces that are familiar to users of client/server and desktop applications, together with the delivery of a distributed Web application. This trend, along with ever-increasing broadband penetration, has permanently changed the Web and how its primary users, businesses and consumers, interact with it.

STATIC VERSUS DYNAMIC WEB CONTENT

Static content is typically text and images that are fixed and rarely updated, served "as is" to any browser that connects to the Web site.

Dynamic content changes based on access requests, usually due to a dynamic program or script that reads and responds to users' requests for data.

Content in RIAs is increasingly dynamic, permitting a level of interactivity and personalization previously not attainable. User-specific information entered into these RIAs is passed back to the Web and storage infrastructure to enable a heightened level of interactivity, furthering personalization and making possible an unprecedented level of customized marketing offers and sales promotions that are tailored and targeted to the individual user's browsing behavior. For instance, consumers can be presented with geographic- and demographic-specific content, content that is tailored to preferences they indicate, or even preferences that are gleaned from their browsing patterns.

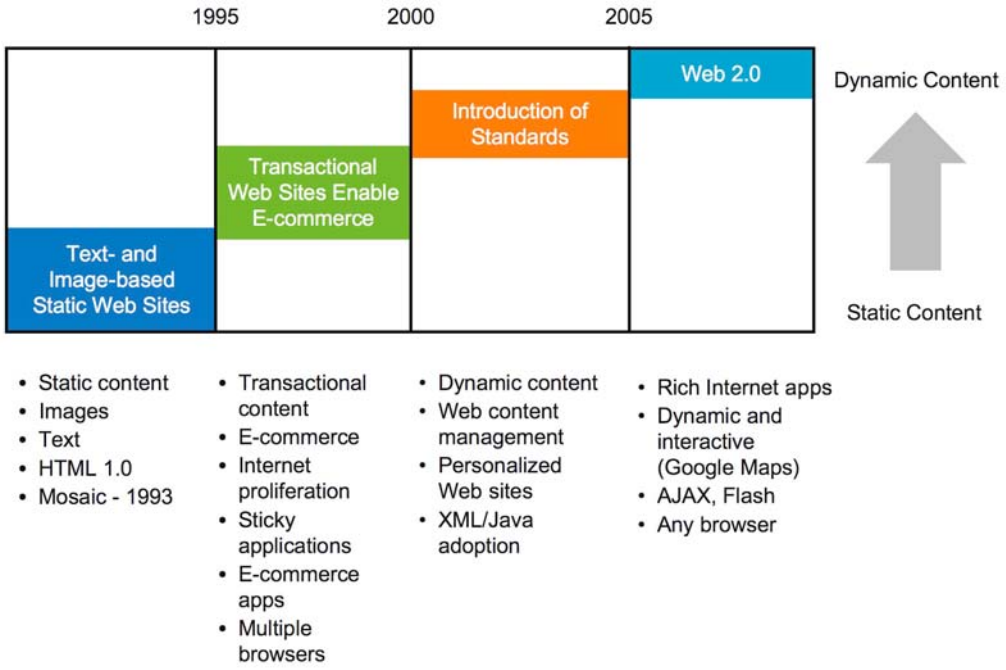

Figure 1) The evolution of Web application technologies.

TECHNOLOGY POWERS WEB 2.0 APPLICATIONS

Most rich Internet Web 2.0 applications are authored using Adobe Flex, Adobe Flash, or AJAX. They share a common architectural framework: When a user first visits the site, the user's browser downloads a sophisticated client-side Internet application engine. As the user interacts with the site, the application or engine communicates with the databases, application servers, and storage at the back-end server to retrieve tiny pieces of data that make up the dynamic content inserted into the page. An early example of a Web 2.0 RIA AJAX-based application is Google Maps. (Other AJAX-based Web 2.0 applications include Flickr, Gmail, and countless others that are rapidly transforming the Webscape.) A user first visiting the Google Maps Web site experiences a slight delay as the map loads. Each subsequent click or panning and zooming of the map results in a call back to the back-end Web server application at the data center to update the AJAX-powered client-side map running in the user's browser.

MOST WEB SITES DON'T DELIVER

It's true that Web sites with dynamic content generated on the fly in real time offer a compelling opportunity to engage with consumers, but dynamic content comes with its own challenges: Dynamically generated content does not load in a Web page as fast as static content does. Also, dynamically generated rich content is much more difficult if not impossible to cache because it's not predetermined what content the user will request. As traffic to a Web site increases, generating pages on the fly for hundreds of thousands of end users simultaneously can have a major impact on the back-end applications server infrastructure, causing major delays in content serving or even complete site failure.

INFRASTRUCTURE UNDER PRESSURE

The dynamic interaction between consumers and the Web infrastructure is a function of the performance, reliability, availability, and scalability of the underlying data storage and compute infrastructure, as well as the unpredictable performance of the Internet itself.

For instance, although AJAX reduces the delay in the interaction between the browser and the display of a page when new data is retrieved from a Web server, an AJAX implementation suffers from the same issues that afflict traditional Web application implementations that use browsers as their user interface. These

issues include performance concerns around scripts in the Web browser and latency/bandwidth issues between the browser and server. For example, AJAX applications may require a number of TCP connections to be open at any one time, which affects server loads and bandwidth requirements due to continuous content refreshes.

Whether the content is delivered with dynamic AJAX, XML, Adobe Flex, or some other sophisticated design or programming language, delivery of this content and applications places greater stress on the Web infrastructure—that is, the compute servers and the networked data storage servers.

Furthermore, most enterprises host their application servers locally, even though their consumer base is distributed nationally or even internationally. Unfortunately, due to Internet problems outside the control of a centrally hosted Web site, the consumer may not be able to access an application or may experience poor performance. These problems are exacerbated when a large number of consumers attempt to access the application simultaneously.

ON THE WEB, THE CONSUMER IS KING

Consumers are very sensitive to the quality and consistency of their experience on a particular Web site or Web service. In the Web 2.0 universe, the consumer is the primary driving force for both content creation and content consumption. This places immense competitive pressure on companies that build their businesses on the web using Web 2.0 frameworks. In the Web marketplace, customer loyalty can shift with a single click. Many businesses believe that they manage their Web sites effectively and meet consumers' needs; however, in survey after survey, industry analysts have concluded that consumers are frequently put off by sites that are excruciatingly slow, are subject to random errors, or that just fail inexplicably.

Slow performance and site or Web application unavailability, whatever the reason, can cause deleterious effects on a business brand, its customer loyalty, and its revenues. Consumer research has found that poorly performing sites suffer damaged reputations from which they may not be able to recover.

2 KEY ASPECTS OF DYNAMIC CONTENT SERVING

Enterprises that are familiar with managing back-end Web storage and compute infrastructure for electronic content delivery networks (ECDNs) have a common misconception that rich dynamically generated content is not cacheable. Although this may be true of certain types of dynamic rich Web content, such as truly random data, it is not true for all such content. For example, the output of relational database stores and application servers is cacheable, as are some of the results of search queries and product catalog queries. This is in general true when the dynamic page content that is rendered on demand is the same or substantially similar for most site visitors, whose queries and requested content can be tracked by unique identifiers known as cookies. A framework that enables caching of dynamic rich content is not only possible but is an important aspect of Web 2.0 application framework for dynamic content generation and delivery.

As discussed earlier, enterprises must meet and maintain very demanding requirements on their Web software and hardware infrastructure solutions to remain competitive. The following sections discuss some of these requirements.

CONTENT CACHING—STATIC VERSUS DYNAMIC

Content fit for caching at the edge of the Internet (closest to the user) can be offloaded from the source content server. Also, a well-designed and globally distributed colocated network of content servers can speed up delivery and also cache content close to the end user, improving the performance of Web sites and therefore improving the end user's experience.

PREDICTIVE PREFETCHING OF CONTENT

A well-designed Web infrastructure solution should be able to predict and prefetch dynamic content to its servers before a visitor actually requests a piece of data or Web content. This is easily understood in the context of Web 2.0 application serving dynamic image content such as Google Maps. In such an application, small pieces of cacheable content that make up the adjoining images that a user may be about to navigate to based on his or her current position on a map can and should be prefetched in order to boost performance. Products such as NetApp FlexCache™ running on the NetApp Data ONTAP® operating system in combination with the NetApp WAFL® file system make this possible by placing disk-based objects into memory ahead of time and delivering them to remote FlexCache storage clients. This makes it possible for the B2C Web 2.0 site to rapidly serve rich, interactive, and dynamically generated page content to customers.

NETWORK OPTIMIZATION

Latency is the key factor that creeps into most configurations that use standard TCP/IP parameters. A well-designed solution should minimize or eliminate the effects of latency. A well-designed solution should also be able to accelerate application performance by minimizing the number of round-trip requests it makes between the end user's browser and the back-end web application infrastructure, to speedily deliver content to the Web page. It should also minimize packet loss, which is inevitable, over the wide area network. Through its FlexShare™ QoS and FlexCache content delivery and application acceleration product suites, NetApp offers a robust and proven solution for content delivery and application acceleration.

NETWORK STORAGE

As discussed earlier, network storage is a critical element of the back-end Web infrastructure. It is also a critical element of the ECDN infrastructure, which handles the optimized delivery of content to end users through a tiered caching model consisting of origin servers and edge servers. (Figure 2 shows a typical FlexCache deployment.) By offloading content that is frequently requested as part of initial access downloads from the storage on the origin servers to the storage on the edge servers of the Internet, the solution can eliminate long and costly origin server accesses over the WAN. This tiered caching approach greatly enhances the performance of Web sites as experienced by end users.

3 NETAPP FLEXCACHE AND FLEXSHARE SOLUTIONS FOR DYNAMIC CONTENT SERVING

The NetApp FlexCache solution, coupled with the unique ability of FlexShare to prioritize I/O and to offer storage QoS capabilities provides unique WAN application performance benefits for ECDN and ISPs for delivering highly interactive Web 2.0 applications to a burgeoning and increasingly discriminating consumer audience. FlexCache is a distributed storage caching and application acceleration solution that uses proven and robust caching algorithms. By placing FlexCache in the edge, CDNs and ISPs can deliver superior user experience and much higher bandwidths. Consumers of Web 2.0 applications are insulated from the high latencies of the WAN that they would experience if they were served directly from a central location.

Figure 2 shows a typical ECDN infrastructure deployed using NetApp.

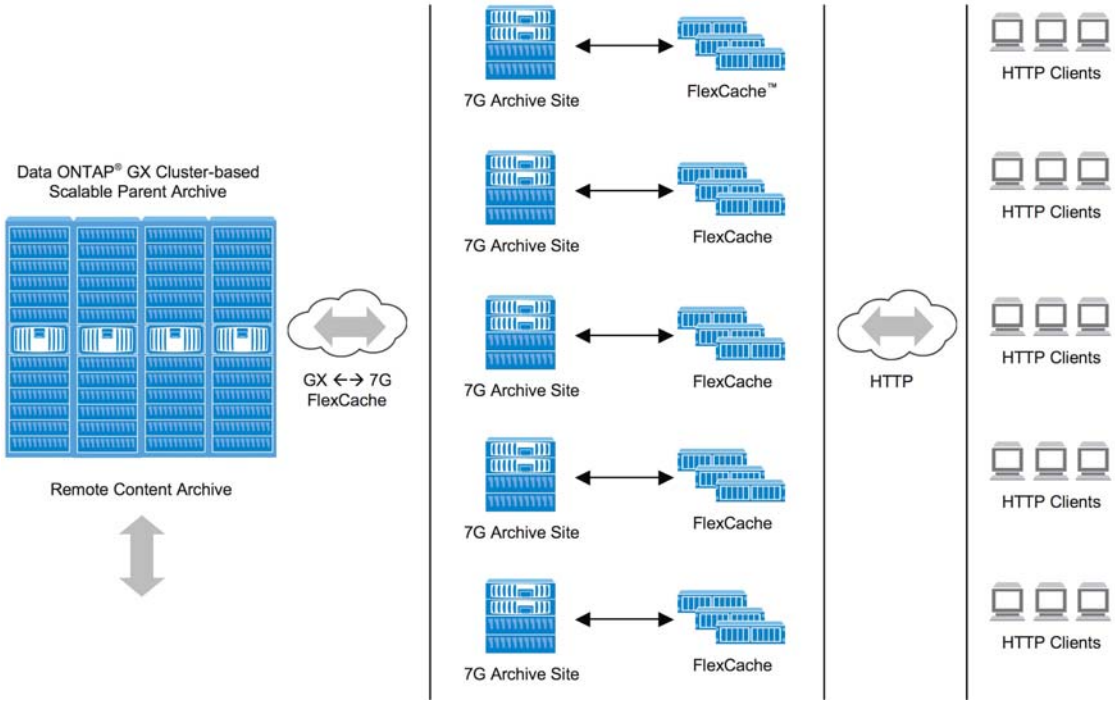

Figure 2) Typical ECDN deployment architecture using NetApp FlexCache for serving Web 2.0 applications on the edge of the network.

HOW NETAPP ACCELERATES AND MANAGES WEB 2.0 APPLICATION DATA

NetApp, the leading global service provider for accelerating content and business processes online, continually evolves its delivery services to keep pace with the evolution of Web sites. NetApp continues to create new services that help enterprises address their 21st-century content and application requirements. NetApp's dynamic storage management and transparent application accelerations solutions address the unique challenges associated with delivering dynamic content, accelerating both download and the interaction between the consumer's browser and the origin site. NetApp's dynamic storage management solutions are built on NetApp Data ONTAP, FAS, FlexCache, and FlexShare technologies. NetApp's proven solutions enable enterprises to extend their B2C Web sites and applications to the Internet edge, bringing content close to consumers. Additional performance improvements are gained through the use of connection and path optimization techniques that dynamically avoid problem spots on the Internet. The result for each Web site is higher availability, superior performance, and greatly increased scalability.

This section describes the benefits of dynamic storage management and transparent storage I/O acceleration solutions.

PERFORMANCE

Dynamic site solutions accelerate the delivery of both static and dynamic, application-generated content, even if it is not cacheable. This includes selecting a fast and reliable path from the Internet edge back to the customer's application server to accelerate the data and calls between the browser and the origin. The solutions can even manage application data calls from the edge of the network (for example, for infrequently changing data that is not time sensitive) and can apply these technologies to dynamic content associated with personalization. Additionally, dynamic site solutions can increase the effective download speed of application content to consumers, as well as accelerate the display of the entire page, regardless of its composition. These solutions dramatically improve the delivery of dynamic content whether compared to the performance of the Web infrastructure or to a traditional CDN.

SCALABILITY

The solution absorbs customers' peak traffic, regardless of the number of site visitors or their location, and eliminates many calls back to the origin. The globally distributed FlexCache platform, along with advanced technology and servers close to consumers, relieves businesses of the need to purchase additional infrastructure. NetApp's advanced storage solutions can scale seamlessly in capacity and performance simply by adding more disk shelves to an existing system or more nodes to an existing NetApp storage cluster, all without any downtime.

AVAILABILITY

Regardless of what type of content a business is serving from its site, the site must be available for consumers to take advantage of it. Unfortunately, Internet outages—beyond the control of any single Web site—are unpredictable and potentially costly in terms of revenues and customer loyalty. Dynamic data protection solutions eliminate site outage problems by leveraging NetApp's strength in site replication and instantaneous zero time, zero space backup and recovery solutions. NetApp also offers a failover option, so that dynamic content can still be delivered even if the origin site is not available.

SECURITY

Because sites and applications are increasingly at risk of outside attacks, businesses must ensure uptime without having to add expensive equipment, software, and personnel, or deploying mirrored sites. NetApp dynamic storage security solutions use protocol-agnostic real-time encryption of data to thwart denial of service attacks, cloaking the customer's origin infrastructure from the public Internet and bypassing bottlenecks to deliver content and applications with the greatest speed and reliability, even in the face of attacks that affect the performance of the entire Internet.

DATA MANAGEMENT

To fully benefit from extending dynamic content and rich Internet applications across a distributed network, a business must be able to control how its content is delivered, cached, published, and routed on the extended infrastructure. NetApp enables customers to configure how their content is cached and served and provides reporting and alerting tools that deliver insight into how the company's extended content and application infrastructure are functioning at all times. NetApp's data management solutions provide these benefits for all content types—static HTML, images, application-generated dynamic content, secure content, personalized content, and software and document downloads—while allowing the enterprise to retain control over how this content is deployed and handled on the NetApp FlexCache platform.

HOW MYSQL POWERS WEB 2.0

The importance of data in the context of Web 2.0 cannot be overstated. An overarching goal of Web 2.0 application design is for the system to harness collective intelligence with network effects so that unique and difficult-to-reproduce data can be generated. At the core of this story is the database that will store and dispense the data. There are many reasons why MySQL is being leveraged time and again by established Web companies and new Web 2.0 companies. The most important reason is cost. Web 2.0 has been summed up as "fail fast, scale fast." The point is that the operational environment is a key differentiator in the success of Web 2.0 adoption, especially in the enterprise. These services are required to fail over and scale transparently without any disruption in service to the consumer. The ability to handle explosive growth in user volume, transactions, and storage capacity is critical to delivering services on a Web-based platform. MySQL addresses these issues via scale out, MySQL Enterprise, and an ecosystem of partner solutions.

SCALE OUT VERSUS SCALE UP

Scale out using MySQL enables organizations to cost-effectively solve database capacity issues that result from increased traffic and transaction volumes. In particular, scale out with MySQL offers these advantages:

- Easily and cost-effectively add capacity to your database infrastructure by using open source software and commodity hardware
- Reduce hardware costs by incrementally adding several low-cost commodity systems versus upgrading high-cost mainframe-class systems
- Reduce software costs and eliminate up-front licensing by scaling out with MySQL
- Improve response time and availability by improving the performance of your system so that users experience fewer interruptions
- Improve scalability by using MySQL replication to distribute large workloads to individual server nodes for execution
- Increase flexibility to right-size the initial purchase of commodity hardware and software and incrementally add capacity

MYSQL REPLICATION

MySQL replication is the key enabler of scale out, discussed in the previous section. Scale out is leveraged extensively by Web 2.0 sites and applications. Replication offers the benefits of reliability, performance, and ease of use. MySQL natively supports one-way, asynchronous replication. Replication works by having one server act as a master, while one or more servers act as slaves. This is in contrast to the synchronous replication that is a characteristic of MySQL Cluster.

Asynchronous data replication means that data is copied from one machine to another, with a resultant delay. This delay can be determined by networking bandwidth, resource availability, or a predetermined time interval set by the administrator. However, with the correct components and tuning, replication can appear to be almost instantaneous to most applications. Synchronous data replication implies that data is committed to one or more machines at the same time, usually by using a “two-phase commit.”

In standard MySQL replication, the master server writes updates to its binary log files and maintains an index of those files in order to keep track of the log rotation. The binary log files serve as a record of updates to be sent to slave servers. When a slave connects to its master, it determines the last position it has read in the logs on its last successful update. The slave then receives any updates that have taken place since that time. The slave subsequently blocks and waits for the master to notify it of new updates.

Figure 3 shows a basic scale out implementation using MySQL replication.

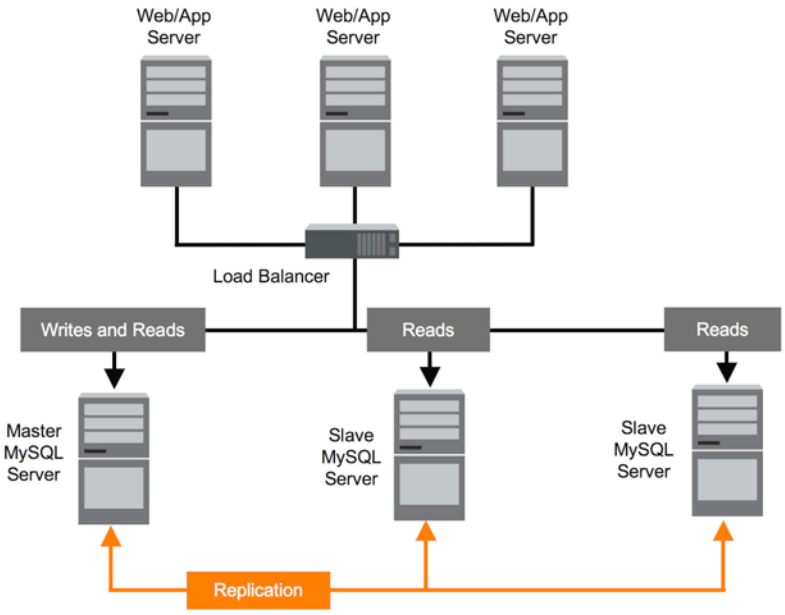

Figure 3) MySQL replication.

MYSQL CLUSTER FOR HIGH AVAILABILITY

Database high availability is increasingly relevant as more business models are based on the premise that the underlying data that drives a company's applications must be consistently available. MySQL Cluster plays a critical role in many of these operations, especially in the area of session management. MySQL Cluster was originally designed to meet the throughput and response time requirements of some of the most demanding enterprise applications in the world. MySQL Cluster can be described as a shared-nothing, synchronous database cluster that supports automatic failover, transactions, and in-memory data storage without any special networking, hardware, or storage requirements. Designing the system in this way allows MySQL Cluster to deliver both high availability and reliability, because single points of failure are eliminated. Any node can fail without affecting the system as a whole. For example, an application can continue executing transactions even though a data node has failed. MySQL Cluster can handle tens of thousands of distributed transactions per second, replicated across data nodes.

Starting with version 5.1, MySQL Cluster supports data storage not only in main memory (RAM) but also on disk. This arrangement allows applications to leverage the benefits of in-memory data storage, which increases the performance of the application and also limits I/O bottlenecks by asynchronously writing transaction logs to disk. And with the introduction of disk-data support, ever larger data sets that do not require the performance characteristics granted to in-memory data can be leveraged within the cluster.

MySQL Cluster automatic node recovery ensures that any failover to another data node contains a consistent set of data. If all of the data nodes fail due to hardware faults, MySQL Cluster ensures that an entire system can be safely recovered in a consistent state by using a combination of checkpoints and log execution. Furthermore, as of version 5.1, MySQL Cluster ensures that systems are available and consistent across geographies by enabling entire clusters to be replicated across regions.

Figure 4 shows a basic cluster architecture.

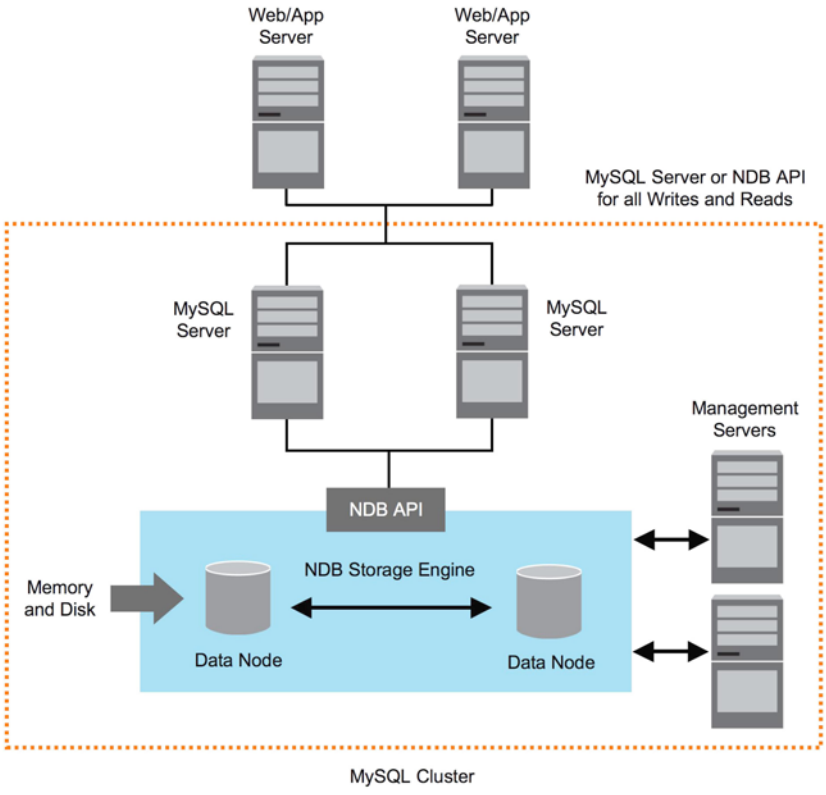

Figure 4) MySQL Cluster architecture.

For more information about MySQL Cluster and how MySQL can be part of your session management architecture, see:

www.mysql.com/products/database/cluster/

www.mysql.com/why-mysql/white-papers/

MYSQL QUERY CACHE

Another feature that is used extensively by Web 2.0 applications is MySQL Query Cache. Accessing application data, query plans, and database metadata in RAM is much faster than repetitively retrieving that same information from disk or building it from scratch.

MySQL Query Cache, introduced in version 4.0.1, delivers excellent gains in the response times of both basic and resource-intensive SQL statements. MySQL Query Cache stores the SELECT queries issued by clients to the MySQL database server. If an identical statement is received, the results are returned from the query cache rather than parsing and executing the statement again.

Other characteristics of MySQL Query Cache include:

- No stale data is ever returned to clients. Data is flushed whenever an UPDATE statement is issued that invalidates the cached data set.
- The query cache is not applicable to server-side prepared statements.
- The expected overhead for enabling the query cache is about 10% to15%.
- Performance gains can be anywhere from 200% to 250% faster when used correctly.

MySQL Query Cache should be enabled in the following scenarios:

- Identical queries are repeatedly issued by the same or multiple clients
- Underlying data being accessed is static or semistatic
- Queries have the potential to be resource intensive or brief, but the result sets are computed in a more complex manner
- Data is presented across many successive Web pages that a user may be navigating through

For more information about MySQL Query Cache, see http://dev.mysql.com/tech-resources/articles/mysql-query-cache.html.

PLUGGABLE STORAGE ENGINE ARCHITECTURE

MySQL's unique Pluggable Storage Engine Architecture (PSEA) gives the developers of Web 2.0 database-driven Web applications the flexibility to choose from a portfolio of purpose-built storage engines that are optimized for specific application domains—OLTP, read-intensive scale out, high-availability clustering, data archiving, data warehousing, and more.

PSEA also provides a standard set of servers, drivers, tools, management, and support services that are leveraged across all the underlying storage engines.

Figure 5 shows how PSEA fits into the overall MySQL Server design.

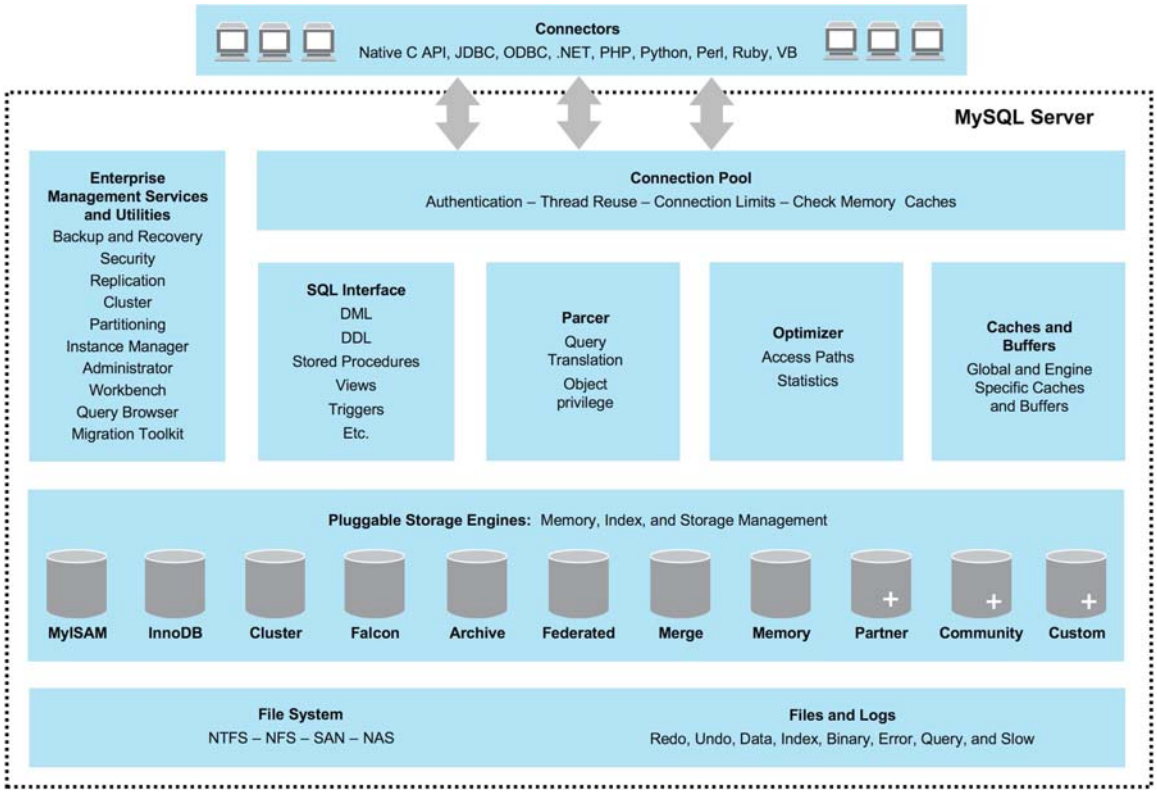

Figure 5) MySQL Pluggable Storage Engine Architecture.

4 TEN REASON TO CHOOSE MYSQL FOR WEB 2.0 APPLICATIONS

SCALABILITY AND FLEXIBILITY

The MySQL database server provides the ultimate in scalability, with the capacity to handle deeply embedded applications with a footprint of only 1MB and to run massive data warehouses holding terabytes of information. Platform flexibility is a feature of MySQL, supporting all flavors of Linux, UNIX®, and Windows®. And the open source nature of MySQL allows complete customization for those who have unique requirements for the database server.

HIGH PERFORMANCE

A unique storage-engine architecture allows database professionals to configure the MySQL database server specifically for particular applications, resulting in amazing performance. Whether the intended application is a high-speed transactional processing system or a high-volume Web site that services a billion queries a day, MySQL can meet the most demanding performance needs of any system. With high-speed load utilities, distinctive memory caches, full text indexes, and other performance-enhancing mechanisms, MySQL offers all the right ammunition for today's critical business systems.

HIGH AVAILABILITY

Rock-solid reliability and constant availability are hallmarks of MySQL; customers relying on it to guarantee around-the-clock uptime. MySQL offers a variety of high-availability options, from high-speed master/slave replication configurations to specialized cluster servers providing instant failover to third-party vendors offering unique high-availability solutions for the MySQL database server.

ROBUST TRANSACTIONAL SUPPORT

MySQL offers one of the most powerful transactional database engines on the market. Features include complete ACID (atomic, consistent, isolated, durable) transaction support, unlimited row-level locking, distributed transaction capability, and multiversion transaction support, so that readers never block writers and vice versa. Full data integrity is also assured through server-enforced referential integrity, specialized transaction isolation levels, and instant deadlock detection.

WEB AND DATA WAREHOUSE STRENGTHS

MySQL is the de facto standard for high-traffic Web sites because of its high-performance query engine, tremendously fast data insert capability, and strong support for specialized Web functions like fast full text searches. These strengths also apply to data warehousing environments, where MySQL scales up into the terabyte range for both single servers and scale-out architectures. Other features like main memory tables, B-tree and hash indexes, and compressed archive tables that reduce storage requirements by up to 80% make MySQL a standout for both Web and business intelligence applications.

STRONG DATA PROTECTION

Because guarding the data assets of corporations is the number one job of database professionals, MySQL offers exceptional security features to ensure data protection. In terms of database authentication, MySQL provides powerful mechanisms for ensuring that only authorized users have entry to the database server, with the ability to block users down to the client machine level. SSH and SSL support are also provided to ensure safe and secure connections. A granular object privilege framework means that users see only the data they should, and powerful data encryption and decryption functions ensure that sensitive data is protected from unauthorized viewing. Finally, backup and recovery utilities provided through MySQL and third-party software vendors allow complete logical and physical backup as well as full and point-in-time recovery.

COMPREHENSIVE APPLICATION DEVELOPMENT

One reason that MySQL is the world's most popular open source database is that it provides comprehensive support for every application development need. In the database, MySQL supports stored procedures, triggers, functions, views, cursors, ANSI-standard SQL, and more. For embedded applications, plug-in libraries are available to embed MySQL database support into nearly any application. MySQL also provides connectors and drivers (ODBC, JDBC, and so on) that allow all forms of applications to make use of MySQL as a preferred data management server. Whether it's PHP, Perl, Java, Visual Basic, or .NET, MySQL offers application developers everything they need to build database-driven information systems.

EASE OF MANAGEMENT

MySQL offers exceptional quick-start capability, with an average time of less than 15 minutes from software download to installation completion. This is true whether the platform is Microsoft® Windows, Linux, Macintosh®, or UNIX. Once installed, self-management features like automatic space expansion, auto-restart, and dynamic configuration changes take much of the burden off database administrators. MySQL also provides a complete suite of graphical management and migration tools that allow the DBA to manage, troubleshoot, and control the operation of many MySQL servers from a single workstation. Many third-party software vendor tools are also available for MySQL to handle tasks ranging from data design and ETL (extract, transform, and load) to complete database administration, job management, and performance monitoring.

OPEN SOURCE FREEDOM AND 24X7 SUPPORT

Many corporations hesitate to fully commit to open source software because they believe they can't get the type of support or professional service safety nets they currently rely on with proprietary software. The question of indemnification comes up often as well. These worries can be put to rest with MySQL; complete around-the-clock support as well as indemnification are available through MySQL Network. MySQL is not a typical open source project, because all the software is owned and supported by MySQL AB, and their cost

and support model provides a unique combination of open source freedom and trusted software with support.

LOWEST TOTAL COST OF OWNERSHIP

By migrating current database-drive applications to MySQL, or by using MySQL for new development projects, corporations are realizing cost savings that may stretch into seven figures. Through the use of the MySQL database server and scale-out architectures that use low-cost commodity hardware, corporations are finding that they can achieve amazing levels of scalability and performance at a cost that is far less than those offered by proprietary and scale-up software vendors. In addition, the reliability and easy maintainability of MySQL mean that database administrators don't waste their time troubleshooting performance and downtime issues, but instead can concentrate on making a positive impact on higher- level tasks that involve the business side of data.

5 MYSQL AND WEB 2.0/LAMP APPLICATIONS ON NETAPP STORAGE INFRASTRUCTURE

As shown in Figure 6, NetApp provides a comprehensive set of data management APIs to interface with you your LAMP-based applications and programmatically access NetApp's data management services on a mature and flexible network storage platform.

LAMP: POWERING THE OPEN ENTERPRISE

Over the past few years, the LAMP stack of open source software components—Linux, the Apache Web server, MySQL database, and the coder's choice of PHP, Python, or Perl—has moved beyond its position as a Web developer's creative toolkit to become a major development platform for the enterprise. Long a cornerstone of the Web's highest volume sites (including Amazon, Friendster, Google, and Yahoo), LAMP has more recently been used to create major online transactional systems for many companies, including Boeing, Disney, Lufthansa, and Sabre.

These bellwethers of online business have discovered that open source software offers the simplest, fastest, most flexible, affordable, and scalable platform for building and deploying Web-based enterprise-class applications. It's a major shift in the software landscape, with implications for the cost, functionality, and performance of enterprise applications.

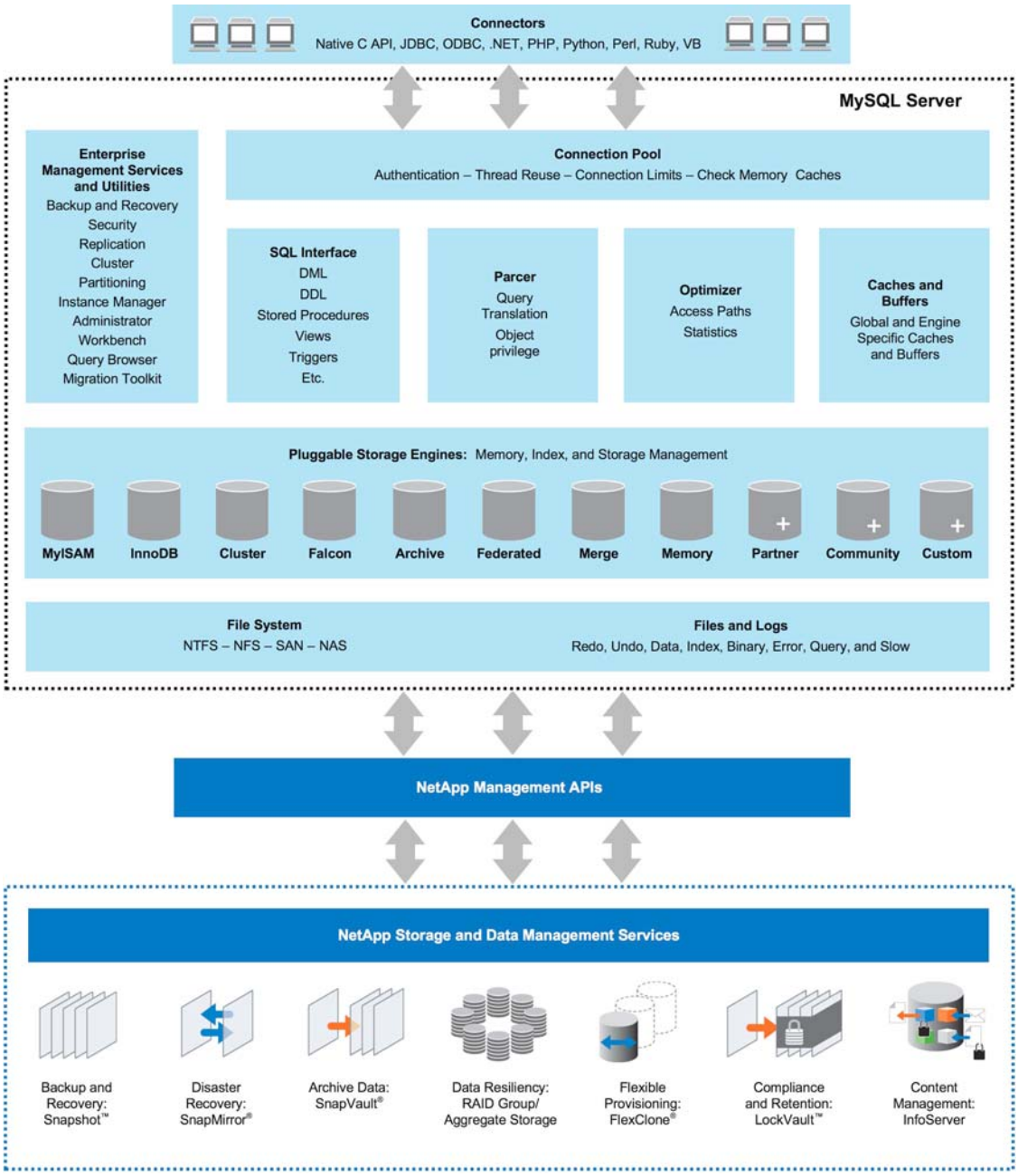

Figure 6) MySQL and NetApp architecture.

If you are thinking of deploying a LAMP stack in your organization, you should consider the NetApp storage platform. NetApp storage provides a comprehensive suite of data management services for LAMP components that is mature and industry proven. NetApp storage offers unique data management, data protection, performance, and security advantages and is fully supported by the NetApp Global Services organization.

LAMP LIGHTS UP THE WEB

LAMP stack components have furnished the software building blocks for Web applications for almost as long as the Web itself has existed. Originally adopted by developers because they were open, free, easily configurable, and robust, these tools evolved quickly as the open source community added features and functionality. Over time, the LAMP stack came to form a de facto platform as tens of thousands of developers polished the integrations and documented best practices.

Individually, the core LAMP components are well supported on NetApp. This section presents a brief overview of the LAMP stack and how it is supported on the NetApp storage and data management platform.

LINUX

Linus Torvald's 1991 announcement that he was coding a free UNIX-type operating system was a watershed event in the open source movement. Since version 1.0 was released in 1994, Linux has become firmly established as an enterprise-class alternative to proprietary UNIX and Microsoft products, largely through the efforts of commercial software providers like Novell that have extended the free Linux kernel and created the support and services infrastructure that is essential to enterprise users.

The Linux operating system is the core of the LAMP stack. All other components are built to run on Linux. All of the most popular Linux OS distributions are certified and proven with the NetApp FAS storage platform, using any storage I/O protocol the customer chooses to deploy. The NetApp FAS platform supports both IP- and block-based storage protocols such NFS, CIFS, iSCSI, and FCP all on the same appliance.

APACHE

The Apache Web server traces its roots to the public domain HTTP daemon developed at the National Center for Supercomputing Applications at the University of Illinois, Urbana-Champaign. In 1995 it was adopted and completely rewritten by a group of volunteers that eventually became the Apache Software Foundation. Apache has been the most popular Web server every year since 1996 and has a current market share of just over 68%, according to a February 2006 Netcraft survey.

The Apache Web server is the de facto standard in the open source LAMP application environment. Apache and its various modules are an integral part of any LAMP-based application.

MYSQL

MySQL is the world's most popular open source database, with more than 8 million active installations. Many of the world's largest organizations—including Sabre Holdings, Cox Communications, The Associated Press, NASA, and Suzuki—realize significant cost savings by using MySQL to power Web sites, business-critical enterprise applications, and packaged software.

MySQL is the standard relational data repository and transactional engine for all LAMP applications. This popular database includes rollback, crash recovery, low-level locking, database replication, clustering, and full-text indexing and search. NetApp's suite of data management software is the storage and data management platform of choice when it comes to the most popular databases in use today, including Oracle®, SQL Server™, Sybase, DB2, and MySQL. NetApp provides a suite of tightly integrated host-side applications to manage these popular database engines. NetApp has partnered with MySQL AB to provide enhanced integration and support services for this popular open source database.

PHP

PHP Hypertext Preprocessor (PHP) is a widely used general-purpose scripting language that is especially well suited for Web development and that can be embedded in HTML. Its relatively simple syntax, ease of use, and open source licensing have made PHP one of the most popular languages on the Web.

PERL

Sometimes called "the duct tape of the Internet," Perl is a cross-platform programming language that is popular with Web developers for its text manipulation capabilities and rapid development cycle. It is highly extensible, with more than 500 third-party modules currently available through the Comprehensive Perl Archive Network (CPAN).

PYTHON

Python is a portable, interpreted, object-oriented programming language developed under the auspices of the Python Software Foundation. It features an elegant but not overly simple syntax, a small number of powerful high-level data types, and a core that can be systematically extended with modules written in a compiled language such as C or C++.

BUSINESS AND TECHNOLOGY TRENDS DRIVE LAMP INTO THE ENTERPRISE

In recent years, a combination of rapid technical advances by open source project teams and commercial pressures on enterprise IT managers have combined to make LAMP a popular enterprise development framework. This section describes some of the factors that are accelerating this trend.

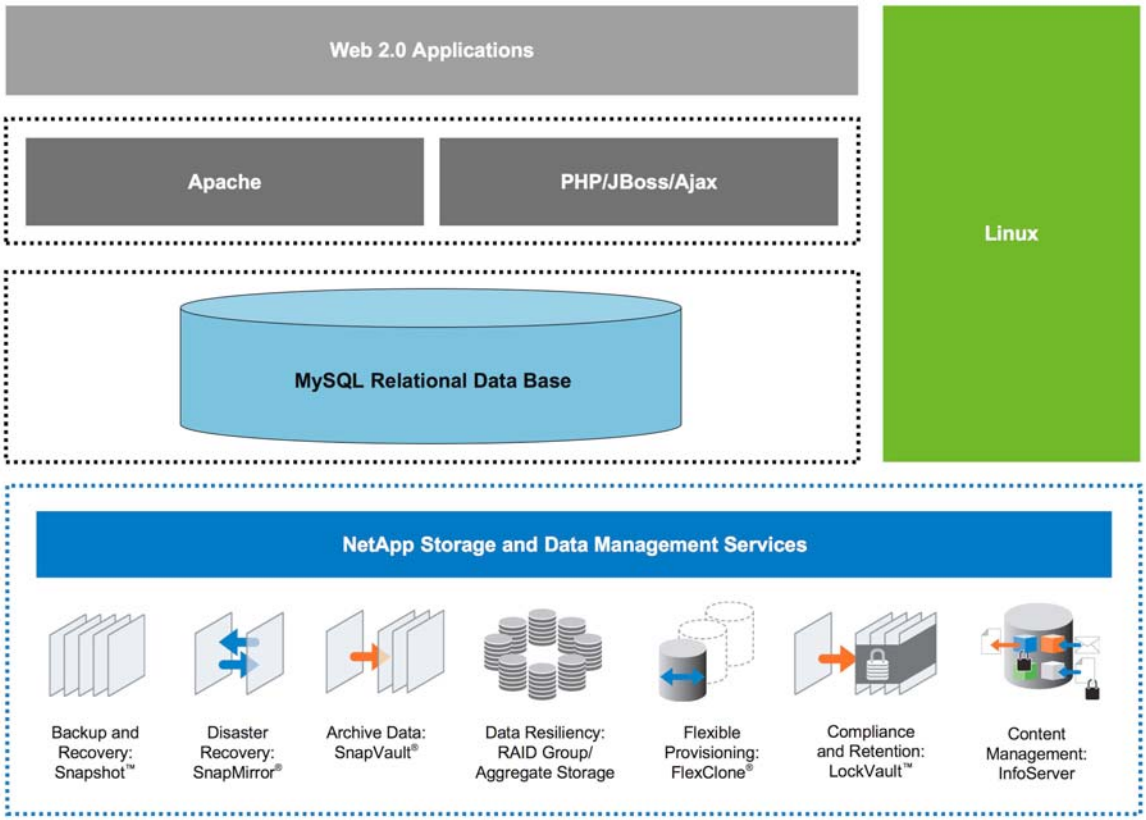

Figure 7) MySQL/LAMP stack on NetApp data management and storage platform.

THE LAMP AND NETAPP OPPORTUNITY: SPEND LESS, ACHIEVE MORE

The LAMP stack, Figure 7, offers a completely open source development stack that is lightweight, inexpensive, highly efficient, and easy to use. Several features distinguish LAMP from proprietary application frameworks and offer advantages that help corporate developers and IT managers bring new applications online more quickly and at lower cost while simplifying their infrastructures and improving their returns on IT investment.

SPEED AND SIMPLICITY

NetApp has always emphasized storage simplicity. That is one key differentiating factor that has made NetApp the leader in protocol-agnostic storage and data management solutions for the enterprise. Through the use of high-level scripting languages, the LAMP stack improves developer productivity. The language components of the LAMP stack— PHP, Python, and Perl—were created specifically to simplify, streamline, and accelerate the type of programming tasks typical of Web development and administration. They are particularly well suited to text handling and to database access for dynamic content generation. In general, they feature simple, clear syntax that makes them easy to learn. They are interpreted rather than compiled,

which simplifies debugging. And they are highly efficient, with a one-line script often performing the same work as many lines of low-level code.

LOW OVERHEAD

The management simplicity of NetApp contributes to tremendous ROI in terms of capital expenditures and operating expenditures. The compact LAMP component stack simplifies deployment and reduces processing overhead. Very tight integration between PHP and Apache, for instance, eliminates the need for application server software and in many instances eliminates an entire physical server tier.

PLATFORM PORTABILITY

The NetApp platform with Data ONTAP OS and WAFL file system is common to all NetApp FAS storage models, offering a unique value proposition for OS interoperability and protocol connectivity. Because LAMP runs on a wide range of hardware platforms, users have maximum flexibility in deployment and server infrastructure design decisions. Of particular value is the option to deploy on clusters or grids of affordable x86-based servers. These utility computing architectures offer an optimized combination of efficient resource utilization, high availability, versatility, and instant scalability.

SECURITY AND STABILITY

The NetApp line of storage security solutions offers an unprecedented level of security for application data, making it less vulnerable than with proprietary development platforms. This is critical in the Web 2.0 and open source environment, to ensure that bugs and security holes can be fixed quickly and easily,

Although Web developers have been using LAMP in very large systems for the better part of a decade, the individual components are not a homogenous software stack that has been vertically integrated by a single vendor. Enterprise developers who simply download free releases of Linux, Apache, MySQL, and PHP can spend considerable time and energy performing and troubleshooting that integration themselves. Documentation and support can be found online, but users may find themselves depending on the good will of volunteers at the worst possible moment. Upgrade releases aren't coordinated among the various component project teams, and new features in one can cause compatibility issues with another. What's more, applications that are built with high-level scripting languages can be difficult to secure, particularly for novice developers.

Developers can take a huge step toward avoiding these and other pitfalls by building on and deploying to a major Linux distribution—one that includes all the essential LAMP components in its distribution bundle and supports each one with a global service organization and complete lifecycle support services: in short, a distribution like NetApp storage.

ABOUT NETAPP

NetApp creates innovative storage and data management solutions that help our customers accelerate business breakthroughs and achieve outstanding cost efficiency. Discover our passion for helping companies around the world go further, faster at www.netapp.com.

ABOUT SUN'S MYSQL DATABASE

MySQL is the most popular open source database software in the world. Many of the world's largest and fastest-growing organizations use MySQL to save time and money powering their high-volume Web sites, critical business systems, and packaged software. At www.mysql.com, Sun provides corporate users with commercial subscriptions and services, and actively supports the large MySQL open source developer community.