Magento®

Magento 2.0

Site Performance and Scalability Optimizations

Table of Contents

Introduction

Magento 2.0 was designed for the next era in eCommerce—to make it faster, easier and more cost effective than ever before to create shopping experiences that are tailored precisely to your business needs.

It all starts with a modern architecture based on popular frameworks and coding patterns that gives you full flexibility to meet virtually any business need with speed and agility. Magento 2.0's modular design reduces application complexity and enables you to easily extend or customize core functionality with off-the-shelf extensions or your own code so you can bring innovative ideas to market fast. A built-in automated testing framework also speeds development and you can easily integrate Magento 2.0 with 3rd party solutions thanks to enhanced and efficient APIs.

Magento 2.0 also offers new functionality to help you create powerful shopping experiences. An all-new, streamlined checkout and enhanced responsive designs provide you with an improved toolset for differentiating your site and growing sales across devices. Redesigned administrative controls and merchandising tools enable more efficient business operations and faster onboarding of new team members so you can deliver better service to your customers. And, with easier upgrades enabled by Magento 2.0's modular architecture, you can immediately take advantage of new functionality as it becomes available.

On top of this flexible and feature-rich new platform, Magento 2.0 also includes comprehensive changes that set a new standard for performance and scalability. Out-of-the-box, Magento 2.0 can now:

- Process significantly more orders per hour on the same hardware
- Offer near-instant server response times for catalog browsing
- Deliver double-digit decreases in response times for cart and checkout pages
- Better handle peak order volume, extra-large catalogs, and outsized customer lists
- Support significantly more simultaneous administrative users on the backend

This paper explores the Magento 2.0 performance and scalability enhancements in depth to help you understand how the new platform can benefit your business and support your long-term growth plans. It also provides configuration recommendations and benchmark testing results that you can use to optimize your Magento 2.0 site performance.

Recommended Magento Configurations

Magento provides a wide range of deployment and configuration options to support different use cases. To optimize performance, many of the settings within Magento must be correctly configured in order to take advantage of the performance and scalability enhancements that are included in Magento 2.0. There are also specific recommended environment configurations that are beyond the scope of this white paper.

IMPORTANT NOTE: All tests referenced in this paper are based on recommended and supported configurations for Magento platforms. Deviation from these configurations may results in varied results, which may not accurately reflect the nature in which the product was designed and architected.

Core Configuration

Magento 2.0 is optimized to run using PHP7 and Varnish. Although it is possible to run with PHP5.6 and without Varnish, **it is not recommended**. It is advised that sites use both PHP7 and Varnish to support the operation of their business and to achieve optimal performance results.

Static Asset Settings

The static assets setting controls how assets such as CSS, JavaScript, HTML, and images are prepared for use by the website.

- Stores->Configuration->Advanced->Developer:
  - Grid Settings: Asynchronous indexing: Enable
  - CSS Settings: Minify CSS Files: Yes
  - JS Settings: Minify JS Files: Yes
  - Merge JavaScript Files: No
  - Template Settings: Minify HTML: Yes
  - JavaScript Settings: Enable JavaScript Bundling: Yes

Page Cache

The page cache setting controls which version of the page cache is used. For the best performance, enabling the Varnish cache is strongly recommended. Note that this requires the deployment and configuration of Varnish.

- Stores->Configuration->Advanced->System->Full page cache
  - Caching Application->Varnish Caching

Email Settings

Email settings control when emails are sent relative to when they are generated.

- Stores->Configuration->Sales->Sales Emails:

    - General settings: Asynchronous sending: Enable

Index Settings

- System->Index Management:

    - All indexers should be in “Update on schedule” mode

Application Cache

- Configure the platform to use Redis in env.php

    - Requires deployment and configuration of Redis

Production Mode

Production mode can either be enabled using a single command:

```
 php bin/magento deploy:mode:set production
```

or can be broken into multiple steps if desired to give developers additional control over the process:

```
php bin/magento setup:di:compile

php bin/magento setup:static-content:deploy

set MODE production manually in web server config
```

Layered Architecture

The Magento 2.0 platform is a layered architecture with four tiers that are optimized for performance and scalability. These are: the client, the page cache, the application, and the database tiers.

This standard web application structure allows for different components of the architecture to be scaled independently in order to meet the requirements of different business use cases and different loads on the system.

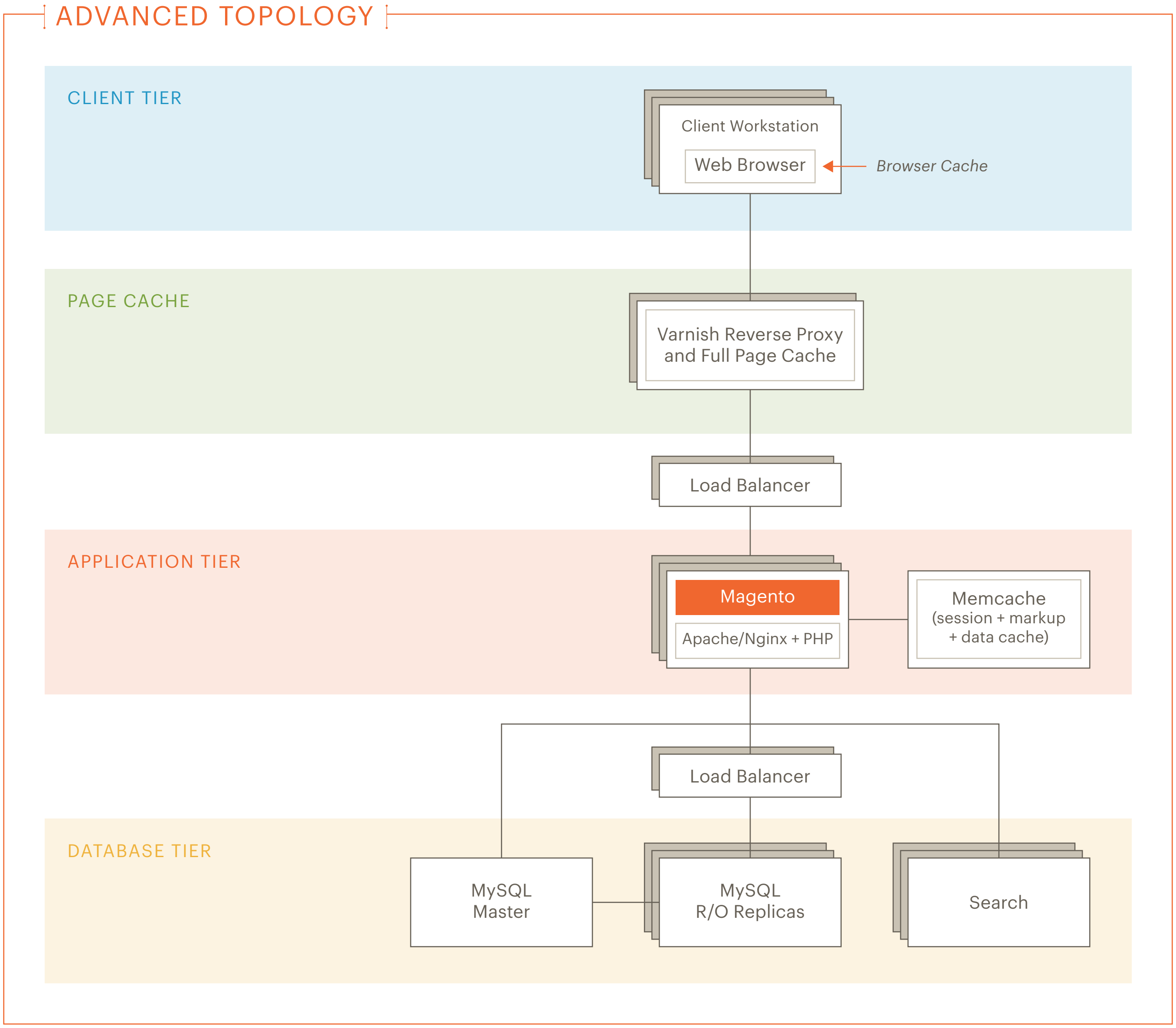


FIGURE 1: Magento Layered Architecture

The following section describes the performance and scalability improvements made in each of these tiers in Magento 2.0.

Performance and Scalability Improvements

Magento 2.0 improves how the application interacts with the browser as well as reduces the amount of data that must be returned from the server to render web pages. These changes significantly improve page response and load times, giving you a much better starting point for developing a fast site.

Client-side Improvements

Page load time is a critical factor for creating a successful eCommerce site and has been shown to directly impact consumer engagement and conversion rates. One key factor in providing a fast page load and good customer experience is reducing the overall page weight or the amount of data that must be loaded to render the page. The amount of data required to render a web page is both a function of the visual design of the page and the technologies and web development strategies used.

Magento 2.0 provides a number of optimizations to reduce page weight and improve response times:

- **Magento 2.0 optimizes the assets required to render the page by using minification**. Minification is a process of removing whitespace, comments, and other unnecessary characters in order to reduce the size of files served to the browser. With Magento 2.0, all HTML, CSS, and JavaScript served by the application are minified to reduce their size. This minification is carried out in advance of requests and is cached so the minified files are available for all website requests. Magento uses industry-standard minification engines such as jShrink for Javascript and cssmin for CSS, as well as an internally-developed engine for HTML. These engines were selected to maximize compression, but merchants can choose to use others if they desire. To further reduce page weight, image files used by Magento websites are compressed using the PHP gd2 library to provide high quality images in the smallest file size.
- **Magento 2.0 maximizes the usage of the browser cache for storing assets**. All JavaScript and CSS are loaded to the browser and cached on the initial web page visited. This minimizes the amount of content that needs to be downloaded to render any subsequent page. To further improve the ability of the cache to service page requests and allow the browser to begin rendering the page, private data (such as consumers' names) are no longer provided as HTML blocks. Now they are returned to the browser as JSON and are inserted into the rendered page when available.
- **Magento 2.0 improves response time by using asynchronous processing during checkout and cart operations**. Items are added to the cart asynchronously to improve response times for the browsing experience. Many checkout operations are also done asynchronously to improve response time, such as address validation and cart updates.

Collectively, these optimizations reduce the amount of data that must be passed to the browser in order to render the webpage. These approaches help ensure that nearly any page design can be efficiently rendered for speed.

Page Caching

Another key to delivering fast page loads is improving the server response time. Magento 2.0 makes extensive use of caching of page content and static assets on the server to accelerate response times.

With Magento 2.0, the application now directly integrates with Varnish page caching out-of-the-box. Varnish is a reverse-HTTP proxy or web accelerator. The Varnish application stores HTML and other files or fragments so that they can be returned extremely quickly.

Varnish provides a very fast and efficient mechanism for serving content that is highly scalable. The requests that are served by the Varnish cache never need to reach the Magento application servers, which reduces the load on the web nodes while dramatically improving the response time. One or more Varnish servers can be used in front of the Magento server(s), an approach that provides a much faster and more efficient system.

Cached content can be composed of different elements with different lifespans or times-to-live using Edge Side Includes (ESI). Edge Side Includes allow for different cached elements to be combined dynamically before being served from the cache. Page elements that cannot be cached, such as cart contents, customer name, or other private data, are provided separately. These page elements are passed to the browser as JSON that can be rendered asynchronously in the browser rather than on the server. This approach allows the page to begin rendering as quickly as possible with the other content added in when it is available, creating a superior user experience. Magento 2.0 handles invalidation of content in the page cache to ensure the right content is served to the website.

Magento 2.0 opens the benefits of Varnish caching to a much broader set of merchants by fully supporting and providing configuration files (.vcl files) for Varnish 3.X and 4.X. These configurations must be updated to reflect the deployment topology, but provide a starting point that makes it significantly easier and less expensive to implement Varnish to improve performance and scalability.

There can be a number of different ways to architect Varnish in the application. The simplest option is to deploy Varnish on the same server that is running Magento. More sophisticated deployment architectures can be used to scale out multiple Varnish servers using a load balancer to distribute traffic between multiple Varnish instances on multiple servers. These deployments can improve scalability and provide redundancy as part of a high availability system.

To use Varnish with secure pages or sites (i.e., served using HTTPS), another application (such as Nginx) must be placed in front of the Varnish server(s) to handle SSL termination.

Magento 2.0 does provide an alternative Page Cache in addition to Varnish. This cache is intended for use in development or in single web node situations where Varnish cannot be used. However, the Page Cache uses a PHP implementation that requires the Magento application to process the caching requests, which causes it to be less efficient and to not yield the same performance benefits. Due to the superior performance and scalability characteristics of Varnish, **it is strongly recommended for use in production deployments**.

Application Enhancements

Magento 2.0 is architected to meet the scalability and stability needs of large and growing merchants and enterprises. The following server-side improvements are implemented in Magento 2.0 to eliminate blocking operations or gridlocks and to improve efficiency of backend business operations:

Asynchronous Order and Product Updates

Large enterprises with lots of orders and product data tend to have multiple Admin users working concurrently on the backend. Magento 2.0 introduces optional asynchronous updates for order management and product data to make concurrent operations efficient and to eliminate gridlock or blocking operations during updates.

For example, when a server is operating at full capacity, an Admin making updates to product descriptions or pick, pack and ship orders is still able to quickly interact with Magento, as the actual processing is queued for later. This means backend Application performance is dramatically improved for situations where there are over 50+ Admin users simultaneously making order updates and over 25+ Admin users simultaneously making product updates.

Job Queue Mechanism

To further advance scalability and responsiveness, additional enhancements were made in our commercial product, Magento Enterprise Edition 2.0. One such Enterprise Edition-only scalability improvement is a job queue based on Rabbit MQ, which allows for asynchronous processing of jobs. Queue workers pull jobs that are placed on the queue when they have capacity. Queue workers can operate using separate server resources from the Magento application servers. This allows the two environments to be optimized separately for their respective loads.

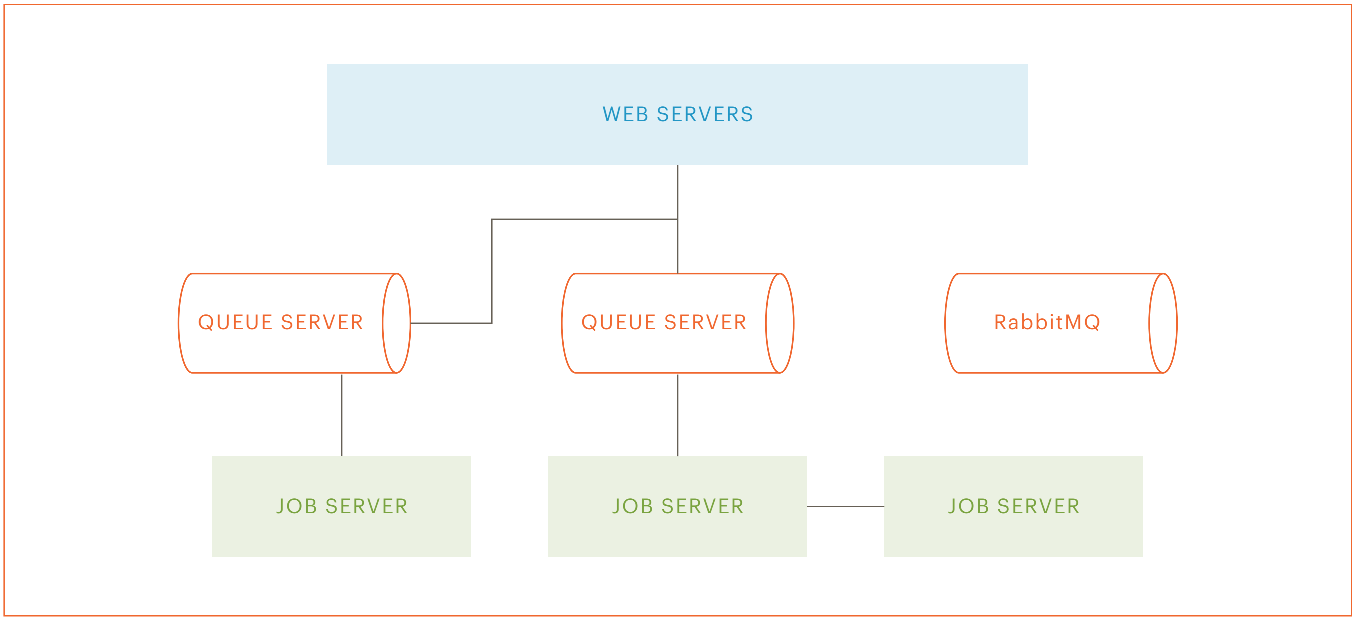


FIGURE 2: Job Queue Architecture

The job queue is provided as a Magento Enterprise Edition 2.0 platform feature that can be customized and extended by the Magento ecosystem for tasks that require highly scalable processing. In subsequent releases, more native features will make use of the job queue for operations such as sending emails, indexing, and asynchronous order insertion.

An example of the job queue's ability to improve efficiency and throughput in operations is deferred stock updates, an optional configuration in Magento Enterprise Edition 2.0. Deferred stock updates allow for inventory levels to update asynchronously as orders are placed to increases checkout throughput. It helps ensure that orders will be captured when checkout transactions are high. This functionality is best suited for high inventory items or products that can easily be backordered. Deferred stock updates can be enabled on either a per-product or per-website scope, giving merchants the flexibility to use where ideal.

PHP Interpreters

Beyond the updates made in the Magento application, there are significant performance advancements that have come from the larger PHP community. One such improvement is PHP7, a radically enhanced PHP interpreter recently released by Zend that has been demonstrated to boost the performance of nearly all PHP applications. Magento recommends and supports the use of PHP7 to achieve optimal performance with the Magento 2.0 platform. Earlier versions of PHP may be used, but may not yield the same performance benefits and are not recommended.

Database Improvements

To further improve scalability in our commercial product, Magento Enterprise Edition 2.0, we made a number of improvements to the database tier of the application. These options allow for tuning and optimization of the databases to better handle high traffic and transaction volumes.

Multiple Database Masters

One major enhancement is the ability to use different databases for different sub-systems or areas of the application. This approach supports up to three different master databases for checkout, orders, and product data that can be broken out in separate database instances. This separation of database instances for different sub-systems effectively shards the Magento database by business entity. Each master database can have multiple slave databases to further scale database read operations.

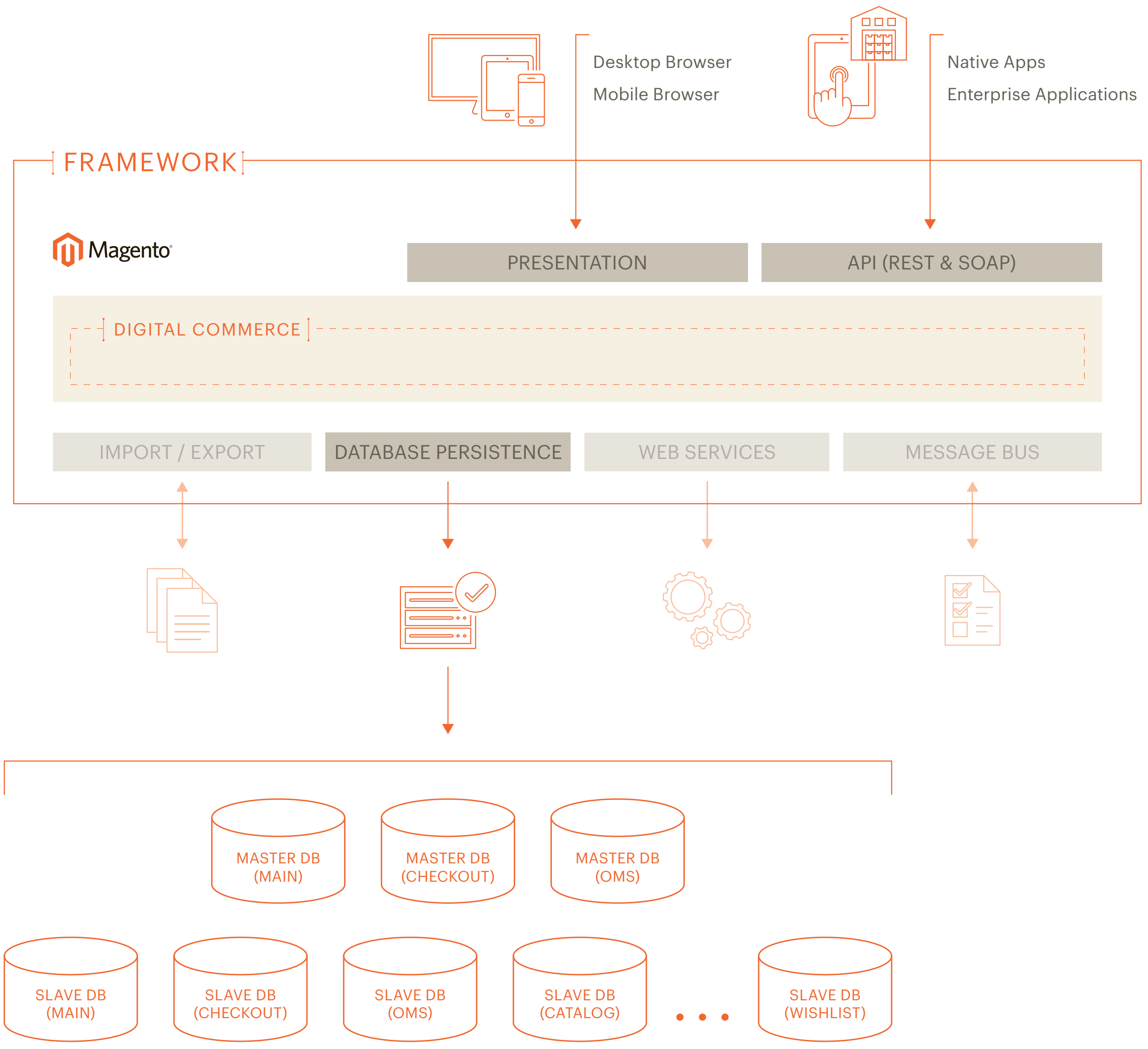


FIGURE 3: Magento Database Tier with Multiple Master Databases

The division helps ensure that the load from merchandising and order management activities can be isolated from users browsing and purchasing on the website. This separation allows different functional areas to be scaled independently depending on the system load and unique needs of the business, such as high order volume in checkout or a very high SKU count.

Magento uses the Command Query Responsibility Segregation (CQRS) database pattern to seamlessly support the routing of queries to the appropriate database. Because of this, developers do not need to know which database configuration is being used, which simplifies customization. CQRS assumes a Create Read Update Delete (CRUD) database access pattern is in use. Our framework uses that assumption to route the generated queries to read (slave) or write (master) databases. Developers customizing Magento do not need to incorporate code to support different database configurations as the framework will handle this automatically.

MySQL Cluster Support

For situations where additional database scalability is needed, Magento Enterprise Edition 2.0 also supports MySQL Cluster for checkout and order management databases. MySQL Cluster is a third-party offering and provides a high-availability, multiple-master solution to scale MySQL. MySQL Cluster manages the sharding of data across multiple database instances. Multiple master databases can be used in each of the application domains as a result. This approach improves the write scalability of the application and enables merchants to use MySQL Cluster to ease management of multiple master databases.

These database improvements represent a major advancement for Magento Enterprise Edition and provide considerable flexibility in how the database tier can be scaled. Now each area of the application can be tuned independently for its expected load. In addition, one can isolate the critical customer facing database interactions from lower priority administrative tasks to achieve comprehensive scale for large and growing enterprise customers.

Performance Toolkit

The Magento Performance Toolkit is a script and a set of JMeter tests that allow for consistent and repeatable performance testing for Magento applications. The script allows the generation of four distinct customer profiles that are intended to represent different eCommerce business sizes. The data from the script populates a Magento instance. The Performance Toolkit is bundled with the Magento distribution (in the setup/performance-toolkit directory).

Performance Results

To show the impact of Magento 2.0's more flexible architecture and performance enhancements, a series of benchmark tests were performed comparing the recommended configurations of Magento Enterprise Edition 2.0 and Magento Enterprise Edition 1.14.2. **These tests revealed that Magento Enterprise Edition 2.0, with its performance enhancements, tight integration with Varnish, and support for PHP7, delivers higher throughput (orders/hour) and faster server response times across the board for both small and large merchant deployments.**

Test Configuration

Software

We tested the recommended and fully-supported out-of-the-box configurations for both products:

- Magento Enterprise Edition 2.0 with PHP7 and Varnish caching
- Magento Enterprise Edition 1.14.2 with PHP5.6 and Full-Page Caching

Testing of other scenarios is not valid as they do not provide data on recommended or supported configurations. Additional software details are available in the Appendix.

Scenarios

Two scenarios were investigated:

- **Small merchant** with $1-$5M in online sales deployed on a single 4-core web node and a single database node. 25 simultaneous JMeter threads (representing 25 concurrent requests) were used to show site performance during a period of peak traffic, such as a sale.
- **Large merchant** with $50-$100M in online sales deployed on five 4-core web nodes and a single database node. 25 to 100 simultaneous JMeter threads (representing 25 to 100 concurrent requests) were used to show site performance under increasing traffic loads.

Both scenarios assumed the following Magento store profile:

Websites / Store Views	SKUs (Simple/ Configurable)	Categories / Nesting	Catalog / Cart Rules	Customers (in Database)	Orders (in Database)
2 / 2	16,000 / 1,000	300 / 3	20 / 20	200	1600

Additional details on the scenarios are provided in the Appendix.

Testing Methodology

The tests emulated typical eCommerce site usage scenarios, such as:

- Catalog browsing, including visiting the home page, a catalog page, a configurable product page, and a simple product page
- Adding both a simple and configurable product to the cart
- Completing the full checkout process (all steps) as a guest and a registered customer

We assumed the following site traffic patterns for user sessions:

- 62% browsing only
- 30% browsing and adding products to the cart, but abandoning
- 4% completing checkout as a guest
- 4% completing checkout as a registered customer

All tests were conducted using the Magento Performance Toolkit and JMeter was used to measure and report server response times in milliseconds.

Small Merchant Deployment

The small merchant test scenario shows that Magento Enterprise Edition 2.0 successfully runs on a small 4-core server, even when handling high loads associated with 25 concurrent users during a sales event. Magento Enterprise Edition 2.0 delivers better performance than Magento Enterprise Edition 1.14.2 across all use cases:

- Processes 28% more order per hour, reaching 597 orders per hour
- Delivers nearly instant response times for catalog pages
- Enables up to 66% faster add-to-cart server response times
- Provides 48% faster guest checkout response times and 36% faster customer checkout response times when all checkout steps are combined

Throughput Results

Orders Per Hour

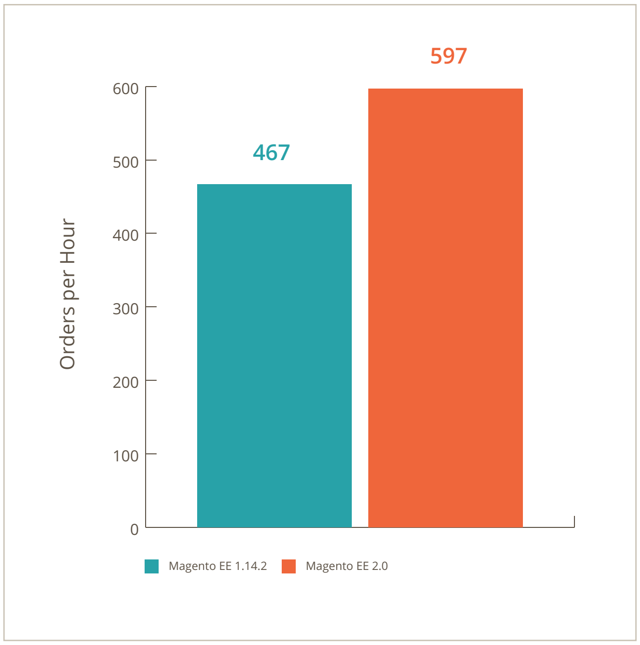


FIGURE 4: Throughput results for 25 JMeter threads show that Magento Enterprise Edition 2.0 can process 597 orders per hour, which is 28% more orders per hour than Magento Enterprise Edition 1.14.2.

Server Response Time Results

Catalog Browsing

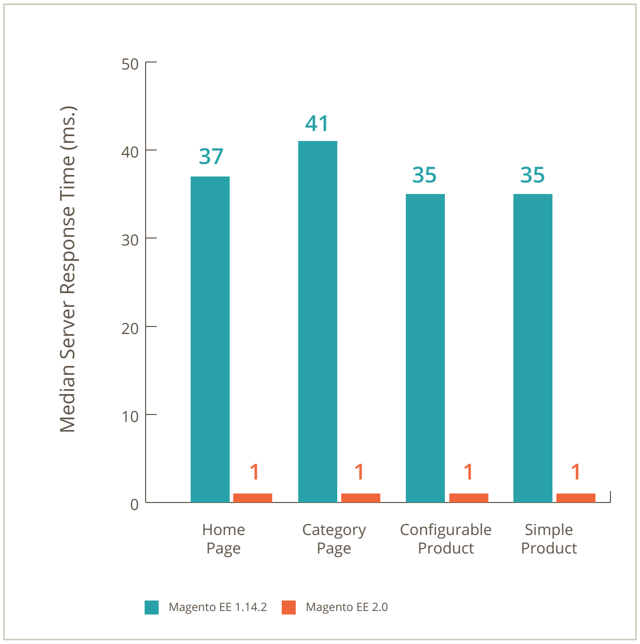


FIGURE 5: Catalog browsing server response times show the Magento Enterprise Edition 2.0 integration with Varnish provides nearly instant responses for cached content. It also generates new pages up to 50% faster than Magento Enterprise Edition 1.14.2 for situation where there is a cache miss or cold cache.

Add to Cart

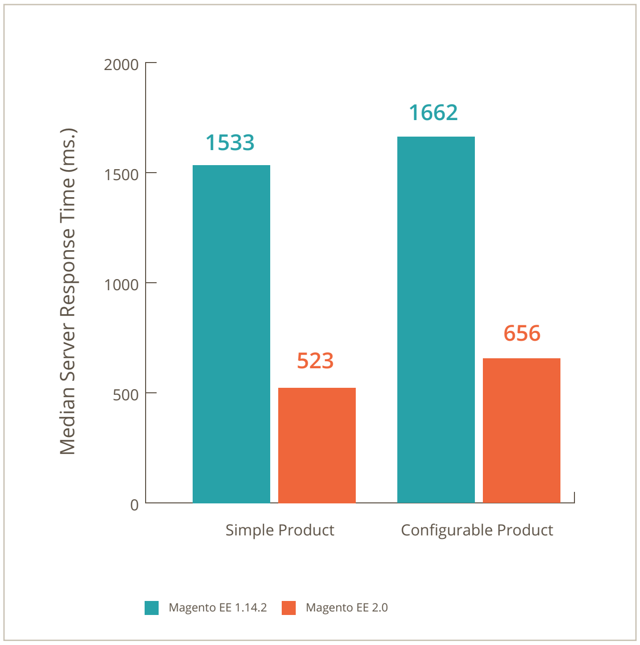


FIGURE 6: Server response times for adding products to the shopping cart are sub-700 milliseconds and up to 66% faster for Magento Enterprise Edition 2.0 when compared to Magento Enterprise Edition 1.14.2.

Customer Checkout Operations (All Steps Combined)

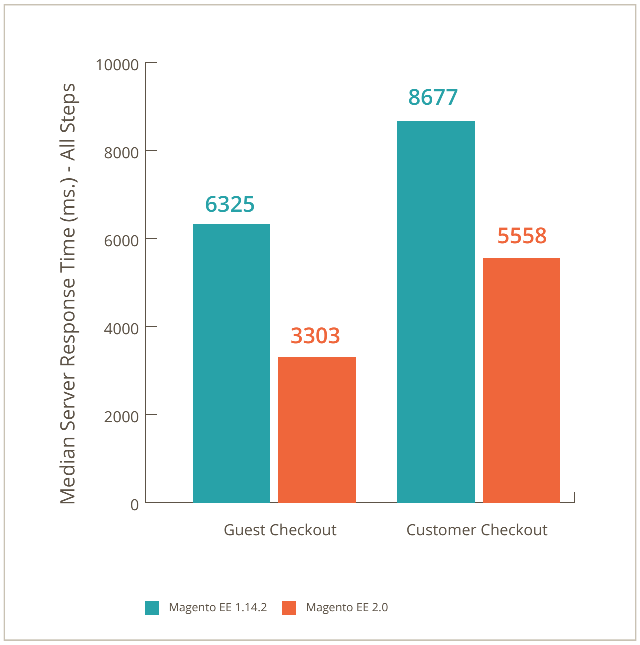


FIGURE 7: Consolidated server response times for all steps of the checkout process show that Guest Checkouts are 48% faster and Customer Checkouts are 36% faster on Magento Enterprise Edition 2.0 when compared to Magento Enterprise Edition 1.14.2.

Large Merchant Deployment

Test results show that Magento Enterprise Edition 2.0 also delivers strong performance for large, high-volume sites, and once again outperforms Magento Enterprise Edition 1.14.2 across all use cases, and under increasing load. Results show that it:

- Processes 39% more order per hour, reaching up to 2,558 orders per hour
- Delivers nearly instant response times for catalog pages
- Enables up to 66% faster add-to-cart server response times that are under 500 milliseconds
- Provides 51% faster guest checkout response times and 36% faster customer checkout response times for all checkout steps combined

Throughput Results

Orders Per Hour

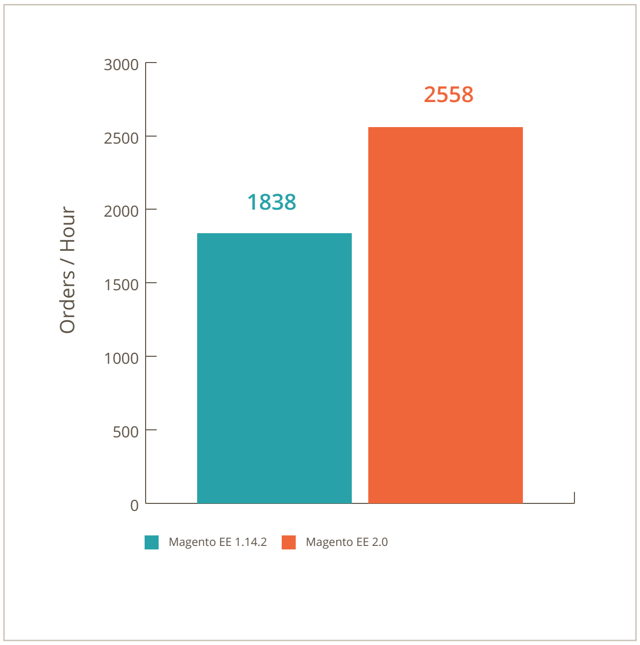


FIGURE 8: Throughput results for 100 JMeter threads show that Magento Enterprise Edition 2.0 can process 2,558 orders per hour, or 39% more orders per hour than Magento Enterprise Edition 1.14.2.

Orders Per Hour - Load Profile

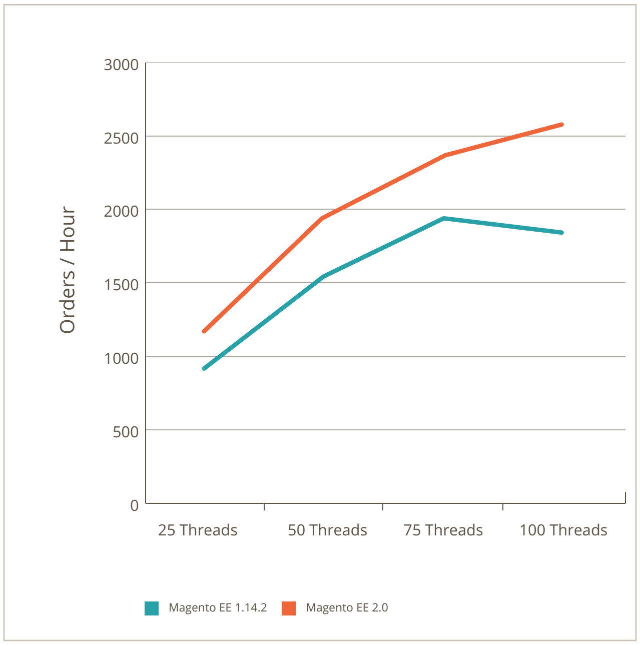
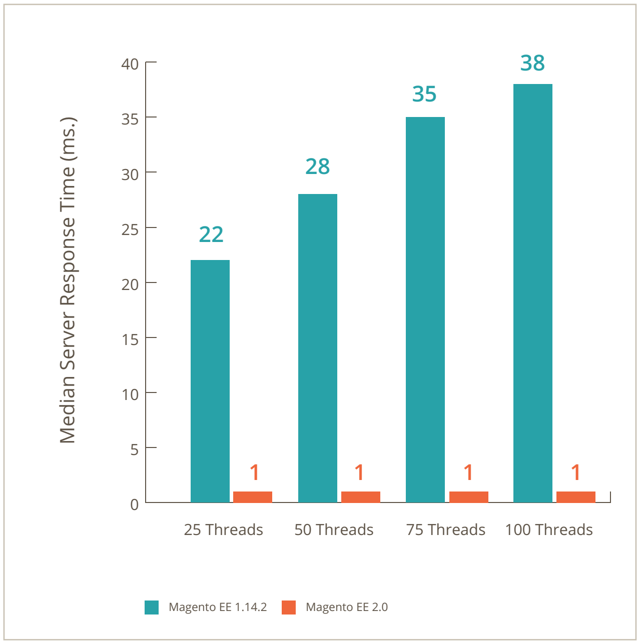


FIGURE 9: Throughput data for a range of JMeter thread counts show Magento Enterprise Edition 2.0 consistently processes more orders per hour than Magento Enterprise Edition 1.14.2 across different load profiles, and is particularly strong at the highest load level.

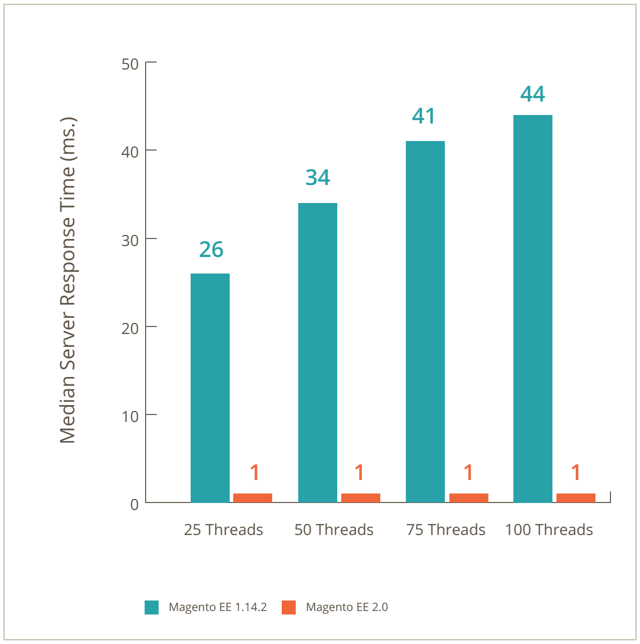
Server Response Time Results

Catalog Browsing

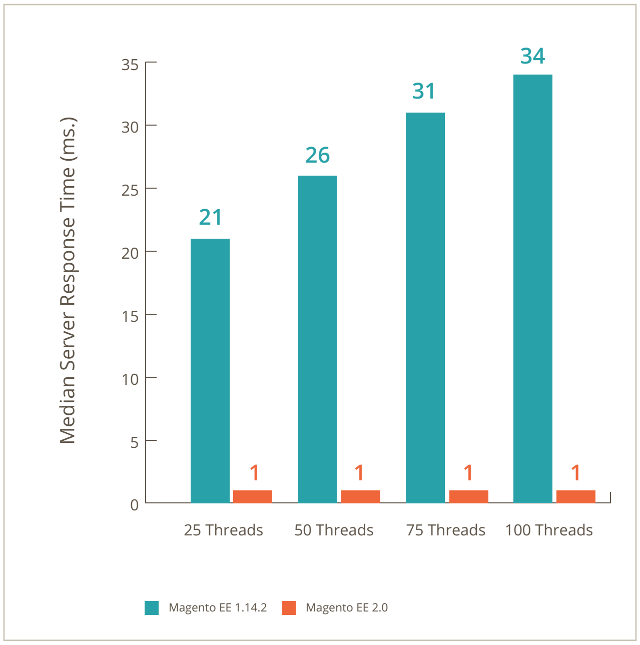
Home Page



Category Page



Configurable Product



Simple Product

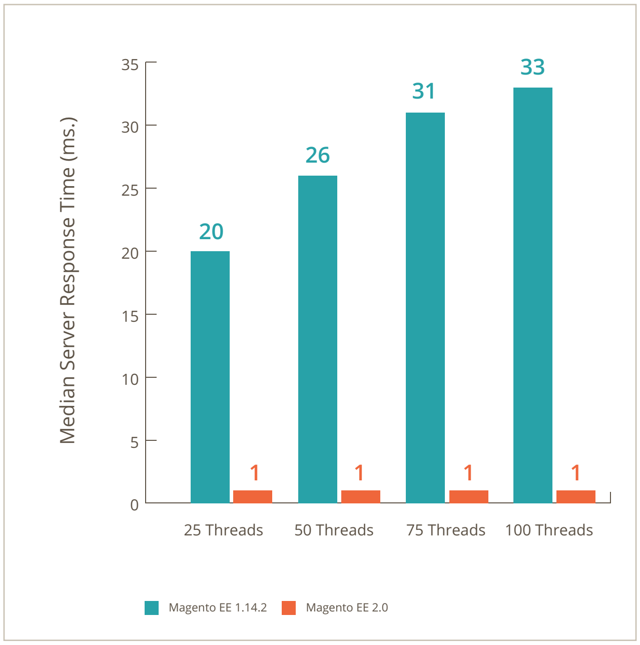


FIGURE 10: Benchmark data for catalog browsing pages show Magento Enterprise Edition 2.0 outperforms Magento Enterprise Edition 1.14.2 and delivers nearly instant server response times across all load profiles.

Add to Cart

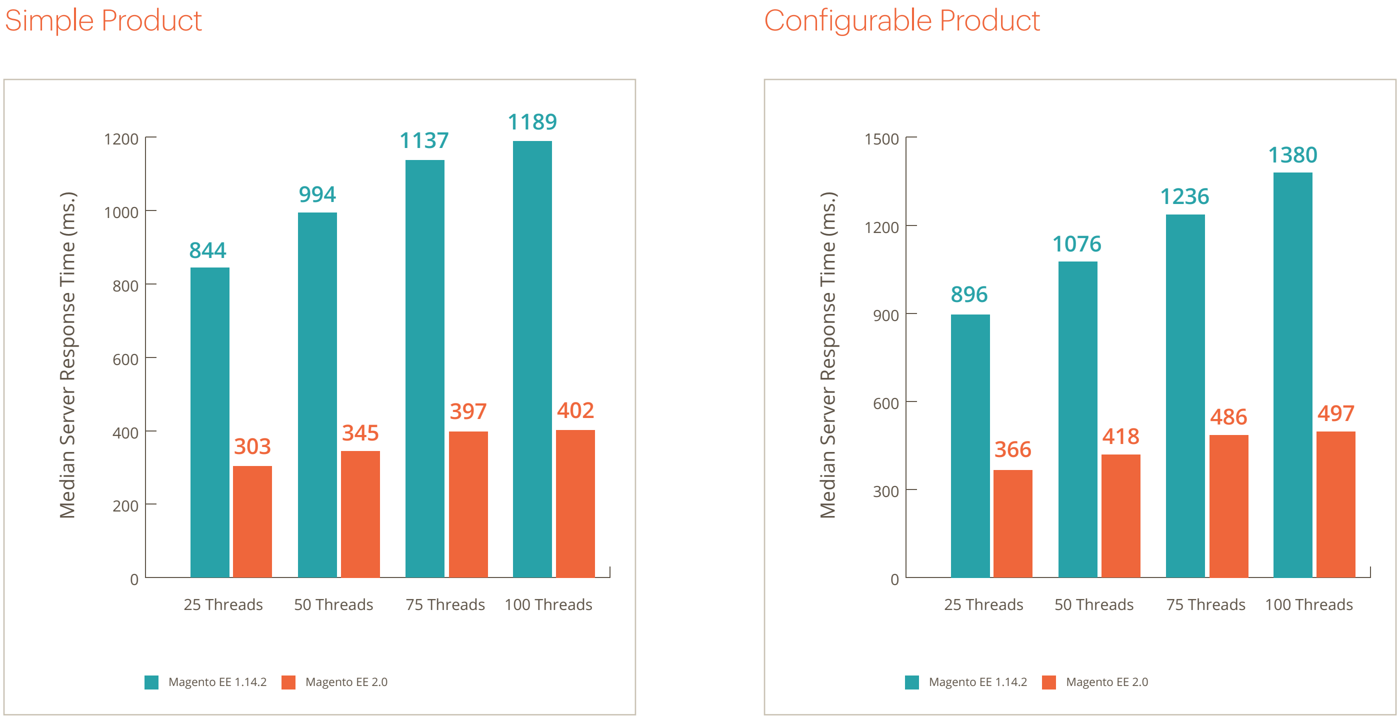


FIGURE 11: Server response times for adding products to the cart on Magento Enterprise Edition 2.0 are up to 66% faster when compared to Magento Enterprise Edition 1.14.2.

Customer Checkout Operations (All Steps)

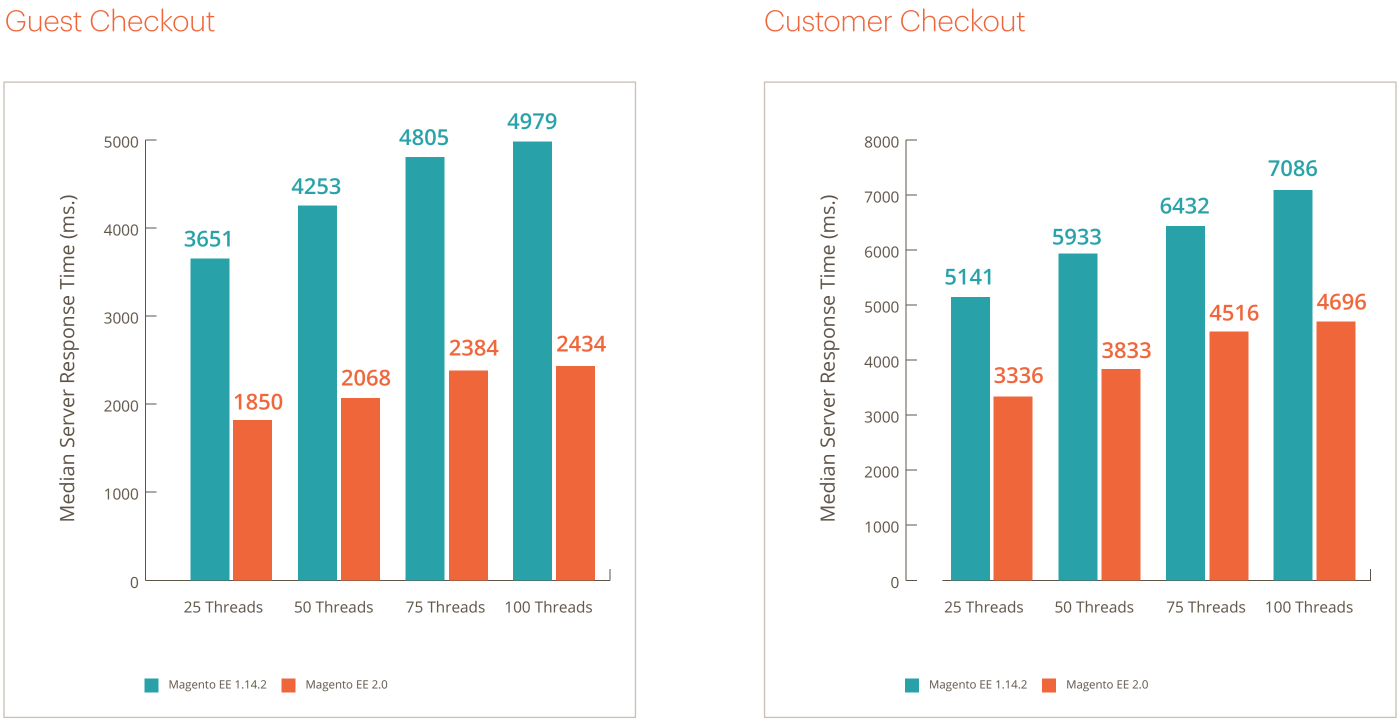


FIGURE 12: Consolidated server response times for all steps of the checkout process show that Guest Checkouts are up to 51% faster and Customer Checkouts are up to 35% faster on Magento Enterprise Edition 2.0 when compared to Magento Enterprise Edition 1.14.2.

Conclusion

Magento 2.0 is a highly efficient eCommerce platform that performs better at scale compared to previous Magento versions. Its top-tier performance and scalability, combined with new functionality, powerful business user tools, and unprecedented flexibility to create tailored shopping experiences make it the next-generation solution that is right for your business.

Powering Magento 2.0's performance and scalability gains are a comprehensive set of enhancements that optimize the client side to reduce page weight, tightly integrate Varnish caching to accelerate page load times, and optimize the application itself to support more efficient job processing and additional concurrent Admin users. It also incorporates new technologies, like PHP7, to boost performance and includes a re-architected database tier with support for multiple master databases to offer a better ability to scale and manage unique system load and business requirements.

Together, these enhancements result in a solution that sets a new standard for performance and scalability. Benchmark tests show that Magento Enterprise Edition 2.0 delivers across-the-board improvements in server response times for both small and large merchants. Catalog browsing, add-to-cart operations, and checkouts are significantly faster and Magento Enterprise Edition 2.0 can process up to 39% more orders per hour, enabling you to get more from your hardware investment. The chart below summarizes all key benchmark test results so you can see for yourself the power of the Magento 2.0 platform.

Small Merchant Deployment

Throughput – orders per hour

Magento Enterprise Edition 1.14.2	Magento Enterprise Edition 2.0
467	597

Server Response Time - milliseconds

	Magento Enterprise Edition 1.14.2	Magento Enterprise Edition 2.0
Catalog Browsing		
Home Page	37	1
Category Page	41	1
Configurable Product	35	1
Simple Product	35	1
Add to Cart		
Simple Product	1,533	523
Configurable Product	1,662	656
Checkout (All Steps)		
Guest Checkout	6,325	3,303
Customer Checkout	8,677	5,558

Large Merchant Deployment

Throughput – orders per hour

Magento Enterprise Edition 1.14.2				Magento Enterprise Edition 2.0			
25 Threads	50 Threads	75 Threads	100 Threads	25 Threads	50 Threads	75 Threads	100 Threads
919	1,531	1,930	1,838	1,179	1,930	2,351	2,558

Server Response Time - milliseconds

	Magento Enterprise Edition 1.14.2				Magento Enterprise Edition 2.0			
	25 Threads	50 Threads	75 Threads	100 Threads	25 Threads	50 Threads	75 Threads	100 Threads
Catalog Browsing								
Home Page	22	28	35	38	1	1	1	1
Category Page	26	34	41	44	1	1	1	1
Configurable Product	21	26	31	34	1	1	1	1
Simple Product	20	26	31	33	1	1	1	1
Add to Cart								
Simple Product	844	994	1,137	1,189	303	345	397	402
Configurable Product	896	1,076	1,236	1,380	366	418	486	497
Checkout (All Steps)								
Guest Checkout	3,651	4,253	4,805	4,979	1,820	2,068	2,384	2,434
Customer Checkout	5,141	5,933	6,432	7,086	3,336	3,833	4,516	4,696

Learn More about Magento 2.0

If you are an existing Magento customer, feel free to reach out to your account manager to learn more about the Magento 2.0 platform. Prospective merchants can speak with a Magento Specialist by calling 1-877-574-5093 or visiting **magento.com/explore/contact-sales**.

Appendix

Merchant Profile Details

Websites / Store Views	SKUs (Simple/ Configurable)	Categories / Nesting	Catalog / Cart Rules	Customers (in Database)	Orders (in Database)
1 / 1	16,000 / 1,000	300 / 3	20 / 20	200	1600

Definitions

- Websites: The number of websites on the store.
- Store views: The number of store views in the system total.
- Categories: The number of categories in the store.
- Nesting: How many nested layers of categories are in the store.
- Catalog / Cart Rules: The number of rules in each category in the system.
- Customers (in DB): The number of registered customers in the system before the test starts.
- Orders (in DB): The number of order existing in the system before the test starts.

Small Merchant Deployment (4 web node cores)

- Software

  - PhP 5.6.13 with Zend Opcache v7.0.6-dev (for Magento Enterprise Edition 1.14.2)
  - PhP 7.0.3-1 with Opcache (for Magento Enterprise Edition 2.0)
  - Nginx 1.6.2
  - MySQL 5.6.28
  - Varnish 4.0.2 (for Magento Enterprise Edition 2.0)
  - Redis 3.0.5
  - CentOS 6.4, CentOS 7.0 and Debian (on core 3.16.7)

- 1 Web Node
  - 4 CPUs Core i7 with hyper threading
  - 8GB of memory
  - 500 GB 7200 RPM SSHD and RAID 1, 5, or 10 disks
  - Running Software
    - Nginx
    - Varnish (for Magento Enterprise Edition 2.0)
    - php-fpm
- 1 Database Node
  - 4 CPUs Core i7 with hyper threading
  - 8GB of memory
  - 500 GB 7200 RPM SSHD and RAID 1, 5, or 10 disks
  - Running Software
    - MySQL
    - Redis (FPC Cache for Magento Enterprise Edition 1.14.2, Session Storage)
- 1 JMeter node (to drive the tests)
  - 4 CPUs Core i7 with hyper threading
  - 8GB of memory
  - 500 GB 7200 RPM SSHD and RAID 1, 5, or 10 disks
- All performance tests leverage real cores. Virtual can bring unexpectedly high deviation.

Larger Merchant Deployment (20 web node cores)

- Software
  - PhP 5.6.13 with Zend Opcache v7.0.6-dev (for Magento Enterprise Edition 1.14.2)
  - PhP 7.0.3-1 with Opcache (for Magento Enterprise Edition 2.0)
  - Nginx 1.6.2
  - MySQL 5.6.28
  - Varnish 4.0.2 (for Magento Enterprise Edition 2.0)
  - Redis 3.0.5
  - Memcache 1.4.21
  - CentOS 6.4, CentOS 7.0 and Debian (on core 3.16.7)

- 5 Web Nodes
  - 4 CPUs Core i7 with hyper threading
  - 8GB of memory
  - 500 GB 7200 RPM SSHD and RAID 1, 5, or 10 disks
  - Running Software
    - Web Node #1:
      - Nginx
      - Varnish (for Magento Enterprise Edition 2.0)
      - php-fpm
    - Web Node #2 - #5:
      - php-fpm

- 1 Database Node
  - 4 CPUs Core i7 with hyper threading
  - 8GB of memory
  - 500 GB 7200 RPM SSHD and RAID 1, 5, or 10 disks
  - Running Software
    - MySQL
    - Redis (Session Storage, FPC storage for Magento Enterprise Edition 1.14.2)

- JMeter node (to drive the tests)
  - 4 CPUs Core i7 with hyper threading
  - 8GB of memory
  - 500 GB 7200 RPM SSHD and RAID 1, 5, or 10 disks

- All performance tests leverage real cores. Virtual can bring unexpectedly high deviation.