

## Table of contents

Enable Fastly IO.....	2
Optimize Fastly request caching .....	3
Request cluster upsize .....	4
Review Support Insights Recommendations (a.k.a SWAT Report).....	4
Monitor traffic with New Relic (Commerce Cloud) .....	4
Upgrade to latest version of Adobe Commerce .....	6
Implement Redis L2 Cache.....	6
Enable MySQL Slave connection.....	8
Review log files in cloud environments.....	8
Monitor disk size growth .....	9
Review slowest database requests.....	9
Install latest ECE-Tools and QPT modules .....	10
Review Apdex score .....	10
Customize New Relic alerts.....	11
Configure cron .....	13
Configure indexers.....	13
Enable asynchronous order data processing.....	15
Enable asynchronous email notifications .....	15
Optimize client-side settings.....	16
Disable and evaluate catalog flat table.....	16
Consider Adobe Commerce on Cloud scaled (split) architecture.....	16

Review this document for tech tips to prepare for peak season. Learn how to prepare your infrastructure for high traffic, prevent downtime, and optimize performance.

## Enable Fastly IO

It is recommended to keep Fastly IO enabled to offload image transformation to the Fastly cloud platform.

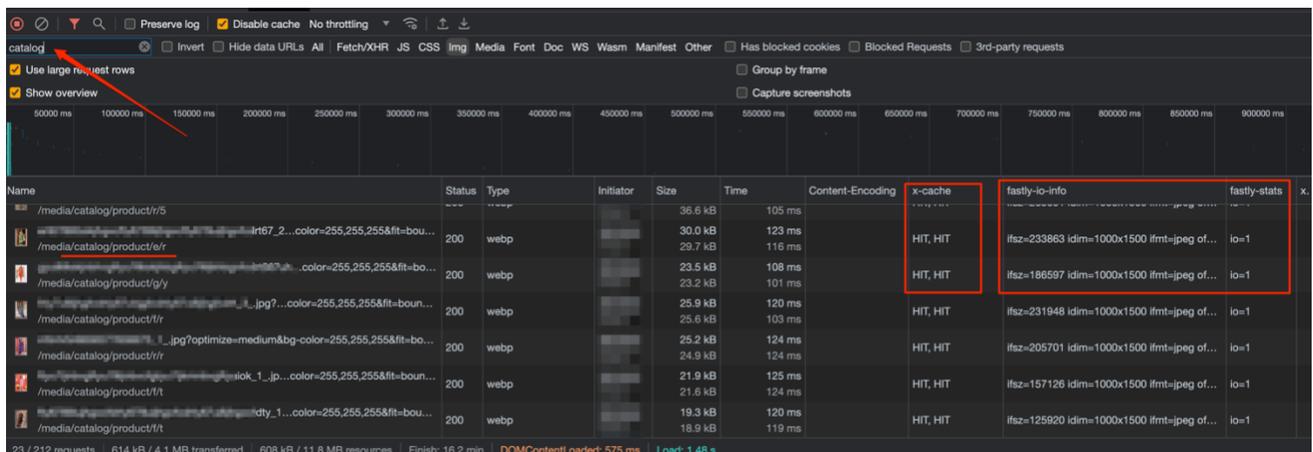
To update Fastly IO:

1. On the Fastly Configuration page in the Default IO config options field, select **Configure**.
2. Review and update Fastly IO snippet is enabled.
3. In the Image Optimization configuration, **Enable deep image optimization** field, and select Yes to turn on deep image optimization.
4. To verify that the correct shield location is configured, refer to the section [Fastly origin shielding](#). To learn more about adjusting backend settings for Fastly performance, refer to [Configure back ends and origin shielding](#) in the Commerce on cloud infrastructure guide.

For more advanced configuration steps, refer to [Fastly image optimization](#) in the Commerce on cloud infrastructure guide.

You can verify if the Fastly image optimization and deep image optimization (for catalog images) is working by looking at the response headers.

- x-cache header in the response returns HIT.
- The fastly-io-info and fastly-stats response headers are populated.
- Image URL path does not use the */cache/* directory.



Name	Status	Type	Initiator	Size	Time	Content-Encoding	x-cache	fastly-io-info	fastly-stats
/media/catalog/product/r/s	200	webp		36.6 kB	105 ms		HIT, HIT		
/media/catalog/product/e/r	200	webp		30.0 kB	123 ms		HIT, HIT	ifsz=233863 idim=1000x1500 ifmt=jpeg of...	io=1
/media/catalog/product/g/y	200	webp		29.7 kB	116 ms		HIT, HIT		
/media/catalog/product/g/y	200	webp		23.5 kB	108 ms		HIT, HIT	ifsz=186597 idim=1000x1500 ifmt=jpeg of...	io=1
/media/catalog/product/f/r	200	webp		23.2 kB	101 ms		HIT, HIT		
/media/catalog/product/f/r	200	webp		25.9 kB	120 ms		HIT, HIT	ifsz=231948 idim=1000x1500 ifmt=jpeg of...	io=1
/media/catalog/product/f/r	200	webp		25.6 kB	103 ms		HIT, HIT		
/media/catalog/product/f/r	200	webp		25.2 kB	124 ms		HIT, HIT	ifsz=205701 idim=1000x1500 ifmt=jpeg of...	io=1
/media/catalog/product/f/r	200	webp		24.9 kB	124 ms		HIT, HIT		
/media/catalog/product/f/t	200	webp		21.9 kB	125 ms		HIT, HIT	ifsz=157126 idim=1000x1500 ifmt=jpeg of...	io=1
/media/catalog/product/f/t	200	webp		21.6 kB	124 ms		HIT, HIT		
/media/catalog/product/f/t	200	webp		19.3 kB	120 ms		HIT, HIT	ifsz=125920 idim=1000x1500 ifmt=jpeg of...	io=1
/media/catalog/product/f/t	200	webp		18.9 kB	119 ms		HIT, HIT		

Please update Fastly IO to help your site be in good health for the Holiday 2023 Peak Season.

## Optimize Fastly request caching

The following checks are important for websites with a default theming approach. But it can be applicable to PWA/Headless websites too.

1. Normalize Unique Parameters: During the holiday seasons, you are likely to run social media-based promotions, including Google Ad campaigns, Facebook, X campaigns, with unique tracking query strings for each customer. Revise the [Fastly Ignored URL Parameters](#) list in the admin panel to Improve cache performance by normalizing these requests.

### ⌵ Advanced Configuration

The screenshot shows the 'Advanced Configuration' section of the Fastly admin panel. It includes three main settings:

- Force TLS:** A toggle switch labeled 'Enable/Disable' with a 'Current state: disabled' indicator. Below it, a note reads: 'Return a 301 Moved Permanently redirect to any unencrypted request, and redirect to the TLS equivalent. Also sets HSTS headers.'
- Admin path timeout (1-600):** A text input field containing the value '181'. To its right is a checkbox labeled 'Use system value' which is unchecked. Below the input is a note: 'Time in seconds for the admin path first byte timeout. Please reupload VCL after making changes.'
- Ignored URL Parameters:** A text input field containing the value 'utm\_\*, gclid, gdftrk\_ga, mc\_\*, trk\_\*, dm\_i\_\*, ke, sc\_\*, fbclid'. To its right is a checked checkbox labeled 'Use system value'. Below the input is a note: 'A comma separated list of ignored query string parameters. Please reupload VCL after making changes.'

2. Ensure your landing pages are cacheable: Check if all the promotion landing pages are cacheable in Fastly. You can check the x-cache header in the response to see if they return any of the following [HIT] or [HIT, HIT] or [HIT, MISS] or [MISS, HIT] for subsequent page loads. For steps, refer to [Debugging](#) on the Fastly [developers] site.
3. GraphQL Query Caching: For PWA/Headless websites, make sure the GraphQL queries are using GET instead of POST whenever applicable.

## Fastly origin shielding

- The purpose of Adobe Commerce Fastly's origin shielding is to reduce traffic directly to the Adobe Commerce origin: when a request is received, a Fastly edge location (Point of Presence) checks for cached content and delivers it. If it is not cached, it continues to the Shield POP to check if it is cached there (if the content has previously been requested even from another global POP, it will be cached). Finally, if not cached on the Shield POP, it will only then proceed to the Origin server.
- Fastly origin shielding can be enabled in Adobe Commerce admin Fastly configuration backend settings. Please choose a shield location which is closest to customer's Adobe Commerce origin data center for the best performance.

- By default, Fastly origin shielding is not enabled.

## Request cluster upsize

The peak season is a crucial time for e-commerce businesses. It brings with it a surge in website traffic, and if you have planned your promotions strategically, you may even be able to anticipate the magnitude of these traffic spikes. However, to ensure a seamless shopping experience for your customers, you might need to consider a temporary cluster upsize, which allows you to allocate more computing resources when handling these traffic surges.

[COMMERCE CLOUD ONLY] - For Commerce Cloud customers, implementing a planned upsize is a straightforward process. All you need to do is raise a support ticket in advance with the date range and the required cluster size. Reach out to your dedicated Account Manager to discuss your current resource consumption and outline your requirements.

**For example:** Assume a Commerce Cloud customer (pro architecture), with daily baseline cluster size of 24 cores (a total of 24 virtual CPUs and 96 GB RAM). If this customer wants to upsize to 96 cores for 7 days for a promotion, the new resources would be roughly 4 times (a total of 96 virtual CPUs and 384 GB of RAM) for just an extra consumption of 504 vCPU days (96x7 – 24x7).

## Review Support Insights Recommendations (a.k.a SWAT Report)

For a more detailed report on your environment, please generate a [SWAT report](#). More information about the SWAT tool can be found in the [Site-Wide Analysis Tool](#) documentation in the Commerce-Operations Tools guide.

For more information on how to prepare for the 2023 Holiday season, please review [The Five P's of Peak Performance](#) blog on the business.adobe.com site. Thank you!

## Monitor traffic with New Relic (Commerce Cloud)

Adobe Commerce Cloud includes a New Relic Observatory system subscription, seamlessly incorporating Fastly logs that stream into New Relic in near real-time. This integration enables you to monitor your traffic patterns and trends in real-time through New Relic, empowering you to take corrective actions.

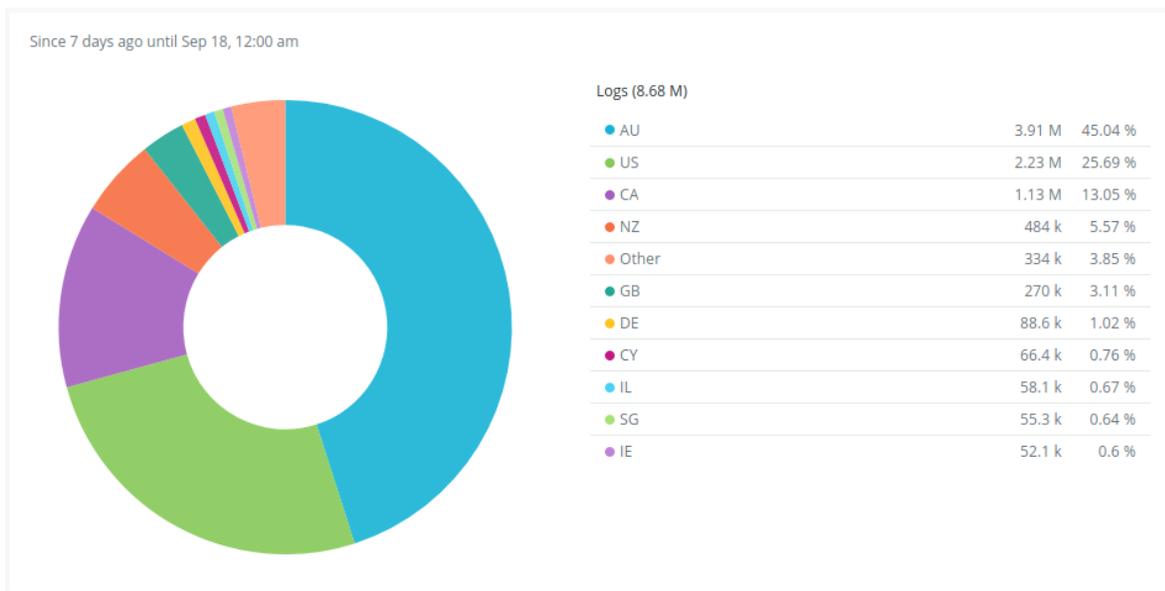
There are many applications of these logs, and how you can utilize them to monitor the system health. You can use these logs to:

- Identify countries your web requests are originated from.

- Find abusive IP addresses/user agents crawling your website.
- Identify malicious web traffic targeting one or more endpoints of your website (e.g., payment).
- Build reports on device types and browser types customers are using.

**For example:** You can monitor the source country of your web traffic to ensure it reflects your promotions/customers' geo locations accurately. Review the following query:

```
SELECT count(*) FROM Log
WHERE cache_status IS NOT NULL
AND project_id = '<YOUR_PROJECT_ID>'
AND (content_type LIKE 'text/html;% ' or url LIKE '%graphql%' or url
like '%rest%')
FACET geo_country_code
SINCE 7 days ago until today
```



You can modify the queries as per your needs, dive deeply into the specific segments and even create dashboards out of them to track them in one place.

## Upgrade to latest version of Adobe Commerce

Please ensure your site is not on an unsupported version of Adobe Commerce, which may impact your site's performance and increase vulnerability to security issues. Please upgrade to the latest version of Adobe Commerce to be secure and ready for the Holiday season.

The [latest release](#) of Adobe Commerce includes many [critical security fixes](#), including enhancements and mitigated issues, that will benefit your project when upgrading from a prior version.

For more information on unsupported versions of Adobe Commerce, please review the [Adobe Commerce Life Cycle Policy](#) in the Commerce-Operations Release Information guide.

For a more detailed report on your environment, please generate a [SWAT report](#). More information about the SWAT tool can be found in the [Site-Wide Analysis Tool](#) documentation in the Commerce-Operations Tools guide.

For more information on how to be prepared for the 2023 Holiday season, please review [The Five P's of Peak Performance](#) blog on the business.adobe.com site.

## Implement Redis L2 Cache

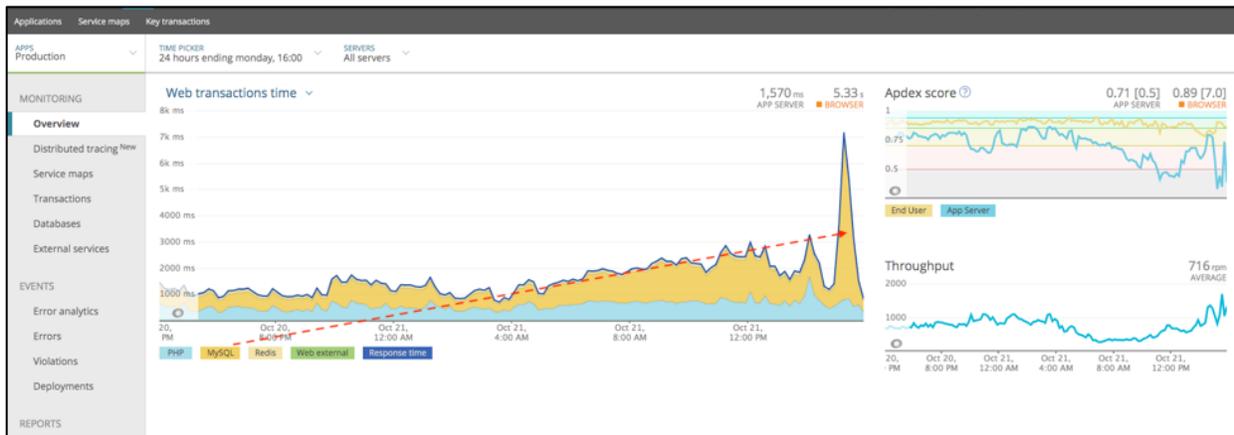
It is important to have the best caching practices implemented for the store to work seamlessly during the peak traffic seasons. Redis L2 cache helps to reduce the network bandwidth to Redis by storing cache data locally on each web node. This can be implemented easily by enabling the deploy variable **REDIS\_BACKEND**. For steps, refer to the [REDIS\\_BACKEND](#) in the Commerce on Cloud Infrastructure Guide.

## Redis Slave Connections

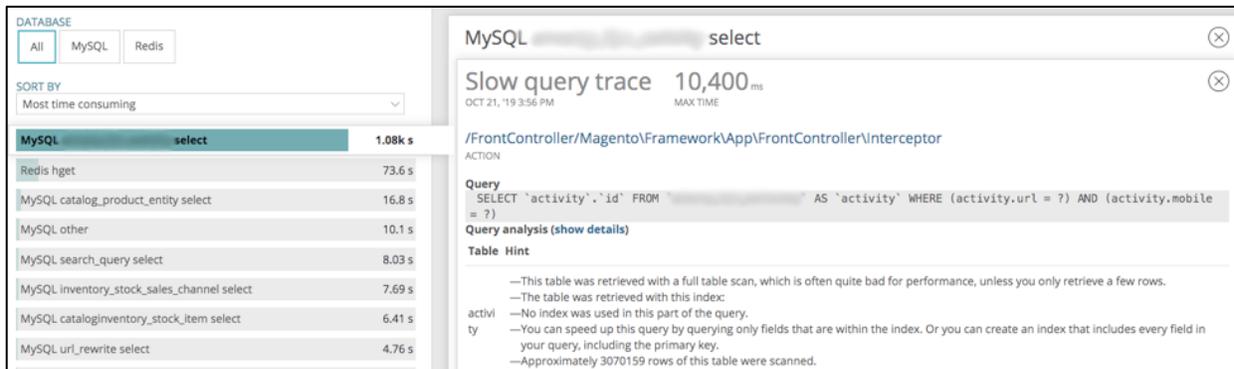
Redis slave connections enable read-only connection to a Redis instance to receive read traffic on a non-master node.

Without Redis Slave connection enabled, MySQL can suffer from high-load bottleneck in Adobe Commerce on cloud infrastructure.

In New Relic APM Overview Chart, check for the first indication that MySQL has become a bottleneck. An example of New Relic data analytics of MySQL performance bottleneck is below:



From here, go to the **Database** tab where the second indication of high throughput or slow SELECT queries in MySQL can be seen. In below sample image, after sorting by **Most time consuming** database transaction, MySQL slow SELECT queries can be observed:



This connection improves performance through load balancing, because only one node handles read-write traffic. Slave connections can be enabled using the deploy variable:

**REDIS\_USE\_SLAVE\_CONNECTION** to true.

For steps, refer to [REDIS USE SLAVE CONNECTION](#) in the Commerce on Cloud Infrastructure Guide.

**Note:** `REDIS_USE_SLAVE_CONNECTION` variable is supported only on Adobe Commerce on cloud infrastructure Staging and Production Pro cluster environments and is not supported on Starter and Scaled architecture projects.

## Enable MySQL Slave connection.

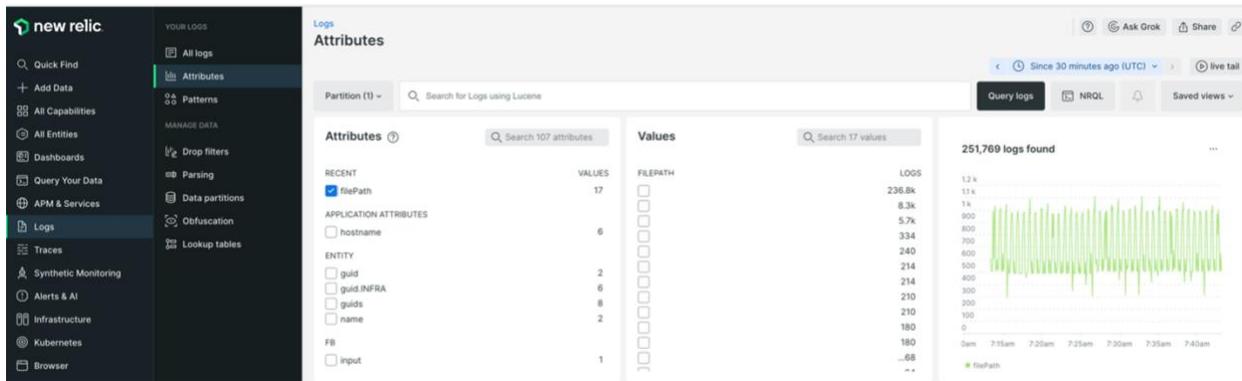
Enable the `MYSQL_USE_SLAVE_CONNECTION` flag in Adobe Commerce Cloud Pro cluster environments. When this setting is activated, Adobe Commerce can direct specific read-only database queries to the slave connection. This will help distribute the load by offloading some query executions from the master connection. For steps, refer to [MYSQL USE SLAVE CONNECTION](#) in the Commerce on Cloud Infrastructure Guide.

## Review log files in cloud environments.

Review the log files in the cloud environments (e.g., Application log files: `~/var/log`) and identify any frequently logged records being written to the default or custom log files. The following are some examples.

1. Review the following default log files and make sure the recurring errors are fixed.
  - a. `~/var/log`
  - b. `~/var/log/exception.log`
  - c. `~/var/log/support_report.log`
  - d. `~/var/log/system.log`
  - e. `~/var/report`
2. Remove debug logs that were previously added for troubleshooting past issues or the identification of exception and error logs that exist in log files (debug logs).

These logs are also available in the **New Relic Logs** section.



## Monitor disk size growth

Adobe Commerce Cloud has two main disk volumes. Monitor these disk volumes to ensure that they have sufficient free space (room to grow) when there is heavy traffic. Adobe Commerce provides a warning when any of these disk volumes reach above 70% usage.

- /mnt/shared (Shared files including logs and media files)
- /data/mysql (Database volume)

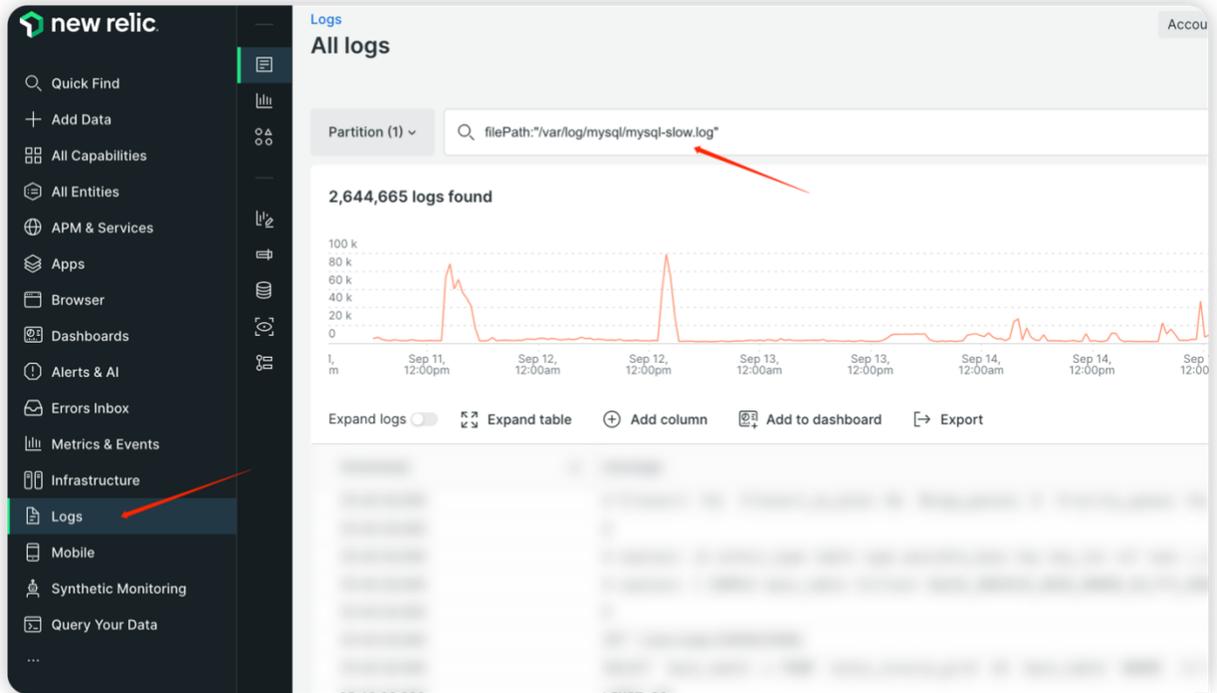
## Review slowest database requests

It is important to regularly monitor and review the most time-consuming database transactions in New Relic. Investigate significantly slow queries/components.

- New Relic > **APM & Services** > Select Environment > **Databases** > **Most time** consuming transactions.

You can also review the mysql-slow.log for slow queries recorded by the system. These logs can be found in New Relic.

- New Relic > **Logs** > Filter by filePath:"/var/log/mysql/mysql-slow.log"



Review the MySQL slow query logs and make sure slow queries are not executed frequently.

## Install latest ECE-Tools and QPT modules

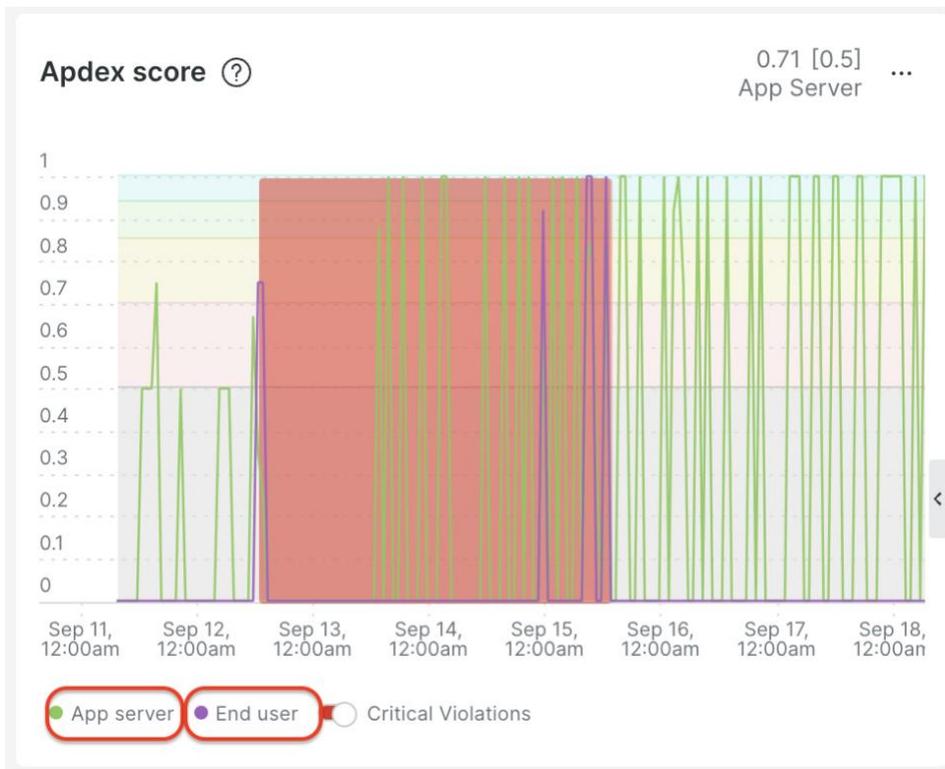
Ensure that the latest ece-tool module and its dependent modules are installed (using the switch **--with-dependencies**) to ensure all the required cloud patches are properly installed according to your Adobe Commerce version. For steps, refer to [Update the ECE-Tools package](#) in Commerce on Cloud Infrastructure Guide.

Review the patch list available in Quality Patch module and ensure the performance patches, which are compatible with the version of Adobe Commerce found in this module, have been applied. Review available patches in [Quality Patches Tool: Search for patches](#) in Commerce-Operations Tools guide.

## Review Apdex score

Apdex score measures users' satisfaction with the response time of your web applications and services. You can review the Apdex score of your Adobe Commerce on cloud infrastructure using New Relic. An Apdex score varies from 0 to 1. A score of 0 is the worst possible score (100% of response times were Frustrated). A score of 1 is the best possible score (100% of

response times were Satisfied). It reports the App server and End user Apdex scores. While the App Server Apdex indicates how your backend is performing, the latter is an indication of your client-side performance.



Along with Apdex, New Relic provides a range of stats to analyze performance issues on Adobe Commerce Cloud. For steps, refer to [Troubleshoot performance using New Relic on Adobe Commerce](#) in the Commerce Knowledge Base.

## Customize New Relic alerts

In addition to the Managed Alerts set by Adobe Commerce Cloud, merchants can set a wide range of alerts and notifications for their platform during peak sale season. This can be anything like notifying the Bot traffic or an increased web response on a GraphQL query.

New Relic Alerts and AI support NRQL based query structures. It can be set up from the New Relic dashboard under **Alerts & AI**.

The screenshot displays the New Relic Alerts & AI interface. On the left is a navigation sidebar with the New Relic logo and various monitoring categories: Quick Find, Add Data, All Capabilities, All Entities, Dashboards, Query Your Data, APM & Services, Logs, Traces, Synthetic Monitoring, and Alerts & AI (highlighted). The main content area is titled 'Alerts & AI' and 'Alert Conditions & Policies'. It features a 'Create new alert condition—query mode' section with a sub-instruction: 'Select a part of your system and set some thresholds. When that signal goes outside your thresholds, we'll create an incident. See our docs'. Below this, there are three radio button options: 'Tell us where to look' (selected), 'Set thresholds', and 'Add details'. To the right of these options is a text input field labeled 'Query the data you want to monitor' containing the word 'SELECT'. At the bottom of the main area, there is a link to 'See our docs' for help with null values, loss of signal, or other query options.

## Configure cron

All asynchronous operations in Commerce are performed using the Linux cron command.

Commerce depends on proper cron job configuration for important system functions, including indexing and queue consumers operation. Failure to set it up properly means that Commerce will not function as expected.

It is critical that Commerce cron is set up and configured correctly using appropriate Unix user in Unix crontab file. Each Unix user has its own crontab file which is the configuration used to run cron jobs for that user.

Note: The script `dev/tools/cron.sh` cannot be executed any longer because it has been removed.

## Configure indexers

- An indexer can run in **Update on Save** or **Update on Schedule** mode.
- The **Update on Save** mode immediately indexes whenever catalog or other data changes.
- **Update on Save** mode:
  - Assumes a low intensity of update and browsing operations in a store.
  - Can lead to significant delays and data unavailability during high loads.
- Recommendation is to use **Update on Schedule** mode in production because it stores information about data updates and performs indexation by portions in the background through a specific cron job.
- The indexer update mode of each Commerce indexer can be set separately on the **System > Tools > Index Management** configuration page.
- The `customer_grid` should not be set to **Update on Schedule** mode.

## Enable asynchronous order placement

- The Async Order module enables asynchronous order placement, which marks the order as received, places the order in a queue, and processes orders from the queue on a first-in-first-out basis. AsyncOrder is disabled by default.
- Use the command-line interface to enable this feature or edit the `app/etc/env.php` file to enable this feature.
  - `bin/magento setup:config:set --checkout-async 1`
- After this configuration is enabled, when an order is received, the order details do not become available right away. This is because the order has not been fully processed yet. It remains in the queue until the `placeOrderProcess` consumer begins, verifies the order with the inventory check feature (enabled by default), and updates the order.
- Before disabling the AsyncOrder module, it is critical to verify that all asynchronous order processes are complete to mitigate existing orders not being processed.

## Enable asynchronous order data processing

- There can be times when intensive sales or promotions occur on storefront. While, at the same time, the Commerce application is performing intensive order processing.
- These two traffic patterns can be distinguished on the database level to avoid conflicts between read and write operations in the corresponding tables.
- To activate this option, navigate to **Stores > Settings > Configuration > Advanced > Developer > Grid Settings > Asynchronous indexing**.
- After this configuration is activated, order data can be stored and indexed asynchronously. Orders are placed in temporary storage and moved in bulk to the Order Management grid without collisions.
- The Asynchronous indexing configuration schedules updates to the following order management grids, and reindexes the data as scheduled by Cron:
  - Orders
  - Invoices
  - Shipments
  - Credit Memos
- By scheduling these tasks, the locks can be avoided that can occur when data is saved and reduces processing time. When enabled, any updates take place only during the scheduled cron job. For best results, Cron should be configured to run once every minute.

## Enable asynchronous email notifications

- Enabling the Asynchronous email notifications setting moves processes that handle checkout and order processing email notifications to the background.
- To enable this feature, go to **Stores > Settings > Configuration > Sales > Sales Emails > General Settings > Asynchronous Sending**.

## Optimize client-side settings

To improve the storefront responsiveness of the Commerce instance, go to **Stores > Settings > Configuration > Advanced > Developer** section, in Default or Developer mode and change the following settings:

Settings Group	Setting	Value
Grid Settings	Asynchronous indexing	Enable
CSS Settings	Minify CSS Files	Yes
JavaScript Settings	Minify JavaScript Files	Yes
JavaScript Settings	Enable JavaScript Bundling	Yes (Not Out-of-the-box)
Template Settings	Minify HTML	Yes

## Disable and evaluate catalog flat table

- The use of flat tables for products and categories is not recommended.
- Use of this deprecated feature can result in performance degradations and indexing issues. Therefore, flat catalog should be disabled via the Adobe Commerce admin, in the Storefront section.
- Some third-party modules and customisations do require flat tables to function correctly. It is recommended that an evaluation be done to understand the impacts and risks associated with having to use flat tables when choosing to utilize those extensions or customizations.

## Consider Adobe Commerce on Cloud scaled (split) architecture

- If after preceding application configurations are applied and code base made more performant, load test results or live infrastructure performance indicate CPU and other system resources are maxed out, then a move to a scaled (split) architecture should be considered.

- By converting to split tier architecture, this changes to a minimum of six nodes: three of which contain ElasticSearch, MariaDB, Redis and other core services; the other three for processing web traffic contain php-fpm and NGINX.
- There are greater scaling possibilities with split tier: core nodes containing databases can be scaled vertically, while web nodes can be scaled horizontally and vertically.
- This allows the expansion of Adobe Commerce Cloud infrastructure on demand for periods of high load activity and on nodes where extra resources are needed.
- If you want to switch to a split-tier architecture due to heavy load expectations for a site, contact your Adobe Account Team.