For more details see how Google Page Speed Optimizer extension works.

Guide for Google Page Speed Optimizer for Magento 2

Improve your store performance with the Google Page Speed Optimizer for Magento 2. Minify code and optimize images without quality loss.

- Automatically reduce images size
- Optimize your database performance
- Minify website code structure
- Improve your webstore pages performance
- Raise your store score in Google PageSpeed Insights up to 99

NEW! Now the extension is compatible with Varnish. You can also add compatibility with 3-rd party extensions which use AJAX.

Find out the general info on how to improve each Google PageSpeed value in our FAQ section.

2 Any concerns about module configuration and usage?

Get an individual training session to explore how the extension works and what practices to implement to achieve the desired result. Save time spent searching for the right settings - discuss configuration cases suitable for your business with the experts and work out the flow that your team needs.

BOOK INDIVIDUAL TRAINING

Magento 2 Google Page Speed Optimizer Pro and Premium extensions are compatible with Hyvä themes. You can find the **amasty/module-page-speed-optimizer-hyva** package for installing in composer suggest (Note: the compatibility is available as part of an active product subscription or Support Subscription).

The Pro and Premium versions of the extension are also compatible with Hyvä Checkout by default.

However, please note that if the 'Hyvä Default (for mobile devices)' option is selected in the general mobile Hyvä Checkout configuration, the image optimization and WebP conversion features will not work for mobile checkout.

Explore our internal **Knowledge Base** to gain even more valuable insights and uncover answers to popular questions about the configuration and features of the Magento 2 Google Page Speed Optimizer extension.

ADD TO CART

GET FREE CONSULTATION

Solution Structure

The solution has 3 feature sets, which are based on the required optimization level.

- Google Page Speed Optimizer Lite: includes lazy load options only
- Google Page Speed Optimizer Pro: includes lazy load, image optimization, code optimization and AMP extension
- Google Page Speed Optimizer Premium: includes all Lite and Pro features + AJAX Shopping Cart and Infinite Scroll extensions

As mentioned above, the PRO version consists of 3 modules: code optimization, image optimization and lazy load. All these extensions should be configured separately. To do this, go to **Stores** \rightarrow **Configuration** \rightarrow **Amasty Extensions**. Start with the **Google Page Speed Optimizer** tab.

Magento 2 Speed Optimization Tips

Before starting the optimization, check your website's current Google score right from the admin panel using the **Diagnostic** option. This built-in testing tool helps you to measure the result after enabling each option. Diagnostics include the tips for Magento optimization based on the current score and the metrics that should be improved. Follow the recommendations to get the most out of the extension. Moreover, for your convenience, Magento performance optimization tips display a precise way for the feature that should be enabled to improve a specific metric.

Configuration 8 Save Config Default Config 👻 Scope: Google Page Speed Optimizer 2.2.0 by amasty a AMASTY EXTENSIONS FEATURE Need help with the settings? Please consult the user guide 8 REQUEST to configure the extension properly. Extensions & Notifications AMP General \odot Google Page Speed Enable Module Yes Ŧ Optimizer [store view] Diagnose your website's speed Image Optimizer Start Diagnostics Request custom optimization Lazy Load

HTML:	\odot
JavaScript:	\odot
CSS:	\odot
Flat Tables:	\odot
Server Push:	\odot
Other Settings:	\odot

Please, mind that the extension automatically gets the URL of a real webstore to check. But if you install the module on a test instance, specify the required URL here:

5/66



You may see the desktop and mobile page speed scores separately. The informative block also contains Google recommendations.

Last update: 2025/07/30 13:37



Note that Google has a limit for checking. If you face the issue below, please wait for some time and restart diagnostics.



Follow the recommendations and check the result once more. You will see the result in the **After** tab.



Before starting configuration, please, set **Magento Production** mode instead of Developer or Default. The Production mode is required to get the best performance of this extension.

Google Page Speed Optimizer

This part of the extension contains code optimization and some other settings to make your webstore load faster.

The Google Page Speed Optimizer extension also works on Magento Cloud, except for the image



optimization functionality.

Using the settings, you can reduce HTML, CSS and JS code size. The extension removes all unnecessary line breaks, tabs, comments and spaces. Also, it includes some advanced settings, such as server push.

Google Page Speed Optimizer 2.1.3 by amasty

FEATURE REQUEST	 Need help with the settings? Please consult the user guide to configure the extension properly.
General	\odot
HTML:	\odot
JavaScript:	\odot
CSS:	\odot
Flat Tables:	\odot
Server Push:	\odot
Other Settings:	\odot

HTML Minification

2025/08/01 13:55 9/66	Guide for Google Page Speed Optimizer f	or Magento 2
HTML:		$\overline{\bigcirc}$
Please note that Mager Default to get the best	nto Production mode should be set instead of Developer or performance of the settings below.	
Minify HTML [store view]	Yes (Recommended) 'bin/magento setup:static-content:deploy' command must be run in the console when any of the settings in the field been changed.	
Prioritize Resource Loading [store view]	Yes Enable to sort the resources in the <head> tag according to best practices.</head>	

Minify HTML - set the option to Yes to enable HTML minification.

Prioritize Resource Loading - when enabled (*Yes*), the extension sorts the resources in the <head> tag, prioritizing the most important ones to ensure faster loading. The order of priority is as follows:

- Meta tags and page title
- Critical resources (inline CSS in <style> and inline scripts)
- External CSS files
- Fonts
- JavaScript

The **Prioritize Resource Loading** feature is available in the Pro and Premium extension versions, as a part of an active support or product subscription. You can find the **amasty/module-optimizersubscription-functionality** package for installation in composer suggest. When any of the settings in the field have been changed, execute the following command:

when any of the settings in the new have been changed, execute the following com

bin/magento setup:static-content:depl oy

JavaScript

In this tab you can create an advanced bundle of JS using **Advanced JS Bundling** feature and JS Minification that will significantly reduce the page size in kilobytes and increase the page load speed. Google rank for both mobile and desktop will be improved.

You will need to use CLI commands to run through the optimization. Please make sure that you have access to the console and know how to do it.

Please create a full backup of your Magento instance and follow the instructions that are displayed in the **Run Optimization** section.

JavaScript:	\odot	
 Please note that Magento to get the best performant 	Production mode should be set instead of Developer or Default ace of the settings below.	
Amasty JS Optimization [store view]	Enabled (Recommended)	
JS Optimization for Magento Cloud [global]	Yes 🔻 🚱	
Run Optimization [global]	Run	
Clear Bundle [global]	Clear	

To run the optimization, set the **Amasty JS Optimization** option to *Enabled*.

Is Magento Cloud - set to Yes if you use Magento Cloud hosting platform.

Click Start.

Easily Clear Bundle if needed.

Automatic Optimization

Step 1. After clicking the **Start** button, the extension starts preparing all the links for optimization.

2025/08/01 13:55	11/66		Guide for Google Page Speed O	ptimizer for Magento 2
-	Page Speed Optimi	zer for		
Advar	nced Javascript C	ptimization		×
	1	•	•	
	Preparing links			
		1. A.A.		
		Please wait		
Extensions &	Notifications			_
Google Page Optimizer	Speed	Enable Module [store view]	:5	•
	Settin	gs		

Step 2. Links processing progress is displayed on the bar.

	amasty	This demo Page Spec	is for the Good	gle for	Go to Extension Pag	je	Explore Den	10
T	Adva	nced Java	script Opti	mization				×
		<u> </u>		2 Processing	links: 2/5	40%	•	
	Extensions 8	& Notifications	General					
	Google Page Optimizer	e Speed		Enable Moo [store	dule view] Yes		•	

When all the links from the bundle are successfully processed, you can add specific pages to JavaScript bundle. Keep in mind that each link should be placed on a new line.

Last update: 2025/07/30 13:37

 $magento_2:google_page_speed_optimizer\ https://amasty.com/docs/doku.php?id=magento_2:google_page_speed_optimizer\ https://amasty.com/docs/doku.php?id=magento_2:google_page_speed_spe$

amasty	This demo is for the Google Page Speed Optimizer for	Go to Extension Page	Explore Demo
Adva	nced Javascript Optimizatio	n V	/e (
You can https://e https://e https://e	add specific pages to javascript bundle (e.g. Ch example.com/checkout/ example.com/contact/ example.com/checkout/ example.com/contact/	eckout page). Every link should	d be on a new line:
			Finish

If any errors are found, the processing will be stopped and the error message will appear. You may check the particular link that caused a failure. For example, 3/5 links has been successfully processed and then the message appears. It means that the 4th link contains some problems. Easily click on it to check if it is valid or not and correct the link or page itself if needed.

amasty	This demo is for the Google Page Speed Optimizer for Magento 2	Go to Extension Page	Explore Demo					
Adva	anced Javascript Optimization		×					
	⊘2		•					
	Processing li	nks: 3/5						
S It can ta Wait unt	click on the link(s) below to open it. ke some time for the page(s) to load. Open all the links til you get the message You can close the window	60% if there are several of them.						
https://google-page-speed-optimizer-m2.magento-demo.amasty.com/training/training-video.html https://google-page-speed-optimizer-m2.magento-demo.amasty.com/gear/bags.html https://google-page-speed-optimizer-m2.magento-demo.amasty.com/women/tops-women/jackets-women.html https://google-page-speed-optimizer-m2.magento- demo.amasty.com/admin/admin/system_config/edit/section/amoptim https://google-page-speed-optimizer-m2.magento-demo.amasty.com/admin/amoptimizer/image/index/								
Next Optimizer	speca o							

Step 3. When optimization is finished, you will see the appropriate message and a list of commands that should be executed.



'Is Magento Cloud' Usage

Previously, the setting 'Is Magento Cloud' was created for Magento Cloud usage. But, to avoid manual deployment, you can also use this setting even if you don't use the Cloud. Follow the steps below to set up an automatic deploy:

1. Set 'Is Magento Cloud' option to 'Yes'

2. Find the **app/etc/config.php** file. There you will see the list of installed modules. To create an optimization file, you will need the following massive:

3. Go back to your admin panel and click **Run Optimization** \rightarrow **Start**.

4. The extension will generate the code that should be put into cloud repository:



Clear Bundle The JS optimization is finished

You can see the detailed instructions for generating in the Advanced Javascript Optimization.

5. Copy the generated code and paste it after 'default' arrow in **app/etc/config.php**:



6. Save the *config.php* file.

Additional JS Options

Include JS of the Admin Area Pages into	No (Recommended)		•
[global]			
Exclude URLs from JS Bundling and Merge [store view]	example.com		
	L Each URL or part of the URL from n	ew line	
Move JavaScript To Page Bottom [store view]	Yes (Recommended)		•
Ignore URL List [store view]	example.com		
	Each URL or part of the URL from n	ew line	
Ignore Scripts that Contain [store view]	data-template="bundle-		li
	Example: <script <b="">data-template=" magento-template"><div><%- data %></div></script> will be excluded.	bundle- option" type= quantity_ %> x <%- d	'text/x- latalabel_
JavaScript Optimization Strategy [store view]	Scripts that Contain	Strategy	Action
	<div [data-role="tocart-form</td" id="map-popup-cli</td><td>Defer 💌</td><td></td></tr><tr><td></td><td>"><td>Async 💌</td><td>Ť</td></div>	Async 💌	Ť
	Add		
	Example: <script <b="">data-template=" magento-template"><div><%- data %></div></script> will be optimized	bundle- option" type= quantity_ %> x <%- d I according to the cho	'text/x- atalabel_ sen strategy.

Include JS of the Admin Area Pages into Merged JS File - enable this option if you want to optimize the admin area as well.

Optimizing the admin area pages sometimes lead to malfunctions and doesn't affect the Google rating.

Exclude URLs from JS Bundling and Merge - specify particular URLs that shouldn't be merged or minified. Start each URL or part of the URL from new line.

Don't forget to run the command

bin/magento setup:static-content:deploy

in the console after any changes in this block of settings have been made.

Move JavaScript To Page Bottom option helps to load all the important content before loading JS. The functionality improves pages loading time.

Ignore URL List - provide URLs of the pages on which JS shouldn't be moved to bottom.

Ignore Scripts that Contain - you may also specify particular scripts and their parts if you want them to exclude from moving.

The sample of *Moving to Footer* feature:



JavaScript Optimization Strategy

Please note that the the ability to choose a strategy for JavaScript optimization is available only as a part of an active support or product subscription. You can find the **amasty/module-optimizer-subscription-functionality** package for installation in composer suggest.

Scripts that Contain - add the scripts you want to optimize according to the chosen strategy.

Example: <script **data-template="bundle-**option" type="text/x-magento-template"><div><%data._quantity_ %> x <%- data._label_ %></div></script>

Strategy - choose the strategy for JavaScript optimization.

- Defer for delayed JS files loading.
- Async to load JS files asynchronously.

If **Amasty JS Optimization** is *Disabled*, you can customize the way of code optimization. Besides Moving JS To Page Bottom, there are 4 additional options to reduce your JS code. You can also exclude particular URLs and scripts.

<u>a</u>masty

JavaScript:

 Please note that Magento Produce performance of the settings below 	ction mode should be set instead of Developer o w.	or Default to get the best
Amasty JS Optimization [store view]	Disabled	• 0
Merge JavaScript Files [global]	Yes	•
	'bin/magento setup:static-content:deploy' command m run in the console when any of the settings in the field changed.	ust be been
Disable JavaScript Merging for Upgrade Your Plan	Desktop	
10	Tablet	
	Mobile	
	The functionality is available in the Premium tariff of th Google Page Speed Optimizer module. To upgrade and functionality please follow the link.	e obtain
Enable JavaScript Bundling	Yes	•
	'bin/magento setup:static-content:deploy' command m run in the console when any of the settings in the field changed.	ust be been
Minify JavaScript Files	Yes	•
4 <u>0</u> ,4	'bin/magento setup:static-content:deploy' command m run in the console when any of the settings in the field changed.	ust be been
Minify JavaScript in PHTML Files	Yes	× 0
(giobal)	The setting will be not performed if the deployment of runs withno-html-minify flag.	statics
Specify Path to Templates to be Ignored [store view]	vendor/magento/module- catalog/view/frontend/templates/product/bre rumbs.phtml	eadc
	Each template from new line. Example: vendor/magento/module-	1

Include JS of the Admin Area Pages into Merged JS File [global]	No (Recommended)	*	?				
Exclude URLs from JS Bundling and Merge [store view]	example.html						
	Each URL or part of the URL from new line	- 1.]				
Move JavaScript To Page Bottom [store view]	Yes (Recommended)	•					
Ignore URL List [store view]	example.html						
	Each URL or part of the URL from new line	1.					
Ignore Scripts that Contain [store view]	data-template="bundle-						
	Example: <script <b="">data-template="bundle-option" type="text/x-magento-template"><div><%- data_quan x <%- datalabel_ %></div></script> will be excluded.	tity_ %>					
JavaScript Optimization Strategy [store view]	Scripts that Contain Strategy A	ction					
	<div [data-role="tocart" asyn(="" id="map-pop Defer 💌</td><td>-</td><td></td></tr><tr><td></td><td>" td="" 🔻<=""><td></td><td></td></div>						
	Add						
	Example: <script <b="">data-template="bundle-option" type="text/x-magento-template"><div><%- dataquant x <%- datalabel_%></div></script> will be optimized according to the chosen strategy.	tity_ %>					

Merge JavaScript Files - select *Yes* to put all JS files into one. As a result, the number of queries will be reduced.

Enable JavaScript Bundling - enable this option to combine all JS files into few bundles and download them for each page. This helps to save time by reducing the number of server requests.

Notice that as browsers download the bundles synchronously, page rendering is blocked until all bundles finish downloading.



Minify JavaScript Files - set to Yes to enable JS minification.

Minify JavaScript in PHTML Files - enable this option to reduce the load time of non-cacheable pages by minifying JS in PHTML files when deploying statics.

The setting will be not performed if the deployment of statics runs with -no-html-minify flag. Also, this setting is recommended to be run **on testing environment first**.

Specify Path to Templates to be Ignored - exclude particular templates from minifying if needed.

Include JS of the Admin Area Pages into Merged JS File - enable this option if you want to optimize the admin area as well.

Optimizing the admin area pages sometimes lead to malfunctions and doesn't affect the Google rating.

Exclude URLs from JS Bundling and Merge - specify particular URLs that shouldn't be merged or minified. Start each URL or part of the URL from new line.

Don't forget to run the command

bin/magento setup:static-content:deploy

in the console after any changes in this block of settings have been made.

This is how the bundling feature works:

Image: Stress of the serve log in the serve log is the serve log in the serve log is the serve log in the serve log is the serv	R D	Ele	emen	ts	Со	nsole	2	Sourc	es	Netv	vork	Pe	erforr	mance	e Me	mory	Ар	plicat	ion	Sec
Itter I Hide data URLS XHR JS CSS Img Media Font Doc WS Manifest me selectbg.svg Ups.html collection.html guery-builder.js captcha.js pressolver.js adapter.js refresh.js authentication-popup.html messages.html captcha.html gallery.png mg03-bro.jpg 207 requests 1.1 MB transferred summit-watch.html summit-watch.html caba05b555d9450782939b255da3e59ca.min.css styles-Lmin.css styles-Lmin.css styles-Lmin.css print.min.css print.min.c			T	Q	N	/iew:	=	$\overline{\gamma_{n}}$		Group	by fra	me		Pres	erve log	I	isable	cach	e	Off
select-bg.svg tips.html collection.html query-builder.js aptchajs messages.js spinner.js resolver.js adapter.js refresh.js authentication-popup.html messages.html gallery.png mg03-br-0.jpg 207 requests 1.1 MB transferred Lements Console Sources Network Performance Memory Application Sec View: T T Coroup by frame Preserve log Disable cache Offi ter Hide data URLs XHR JS CSS Img Media Font Doc WS Manifest me summit-watch.html e3ba03b55d9450782339b255da3e59ca.min.css styles.css logo.svg loader1.gif BS020287c49c5469f4a7565c494173.min.js Luma-tons.woff2 fonts_e3ba05b5d9450782339b255da3e59ca.min.css print.min.css pensans-400.woff2 opensans-300.woff2 galery.png mg03-br-0.jpg	ilter						Hide	data	URLs	All	XHR	JS	CSS	Img	Media	Font	Doc	WS	Ma	nifest
select-bg.svg tips.html collection.html query-builder,is messages.js spinner,is resolver,is adapter,is refresh.js authentication-popup.html messages.html gallery.png mg03-br-0.jpg 207 requests 1.1 MB transferred Console Sources Network Performance Memory Application Sec Console Sources Network Performance Memory Application Sec Sources Network Performance Memory Application Sec Styles.css Isoposay Ibader-1.gif Sto2028c7c49cc5489f4a756sc494173.min.js Luma-tcons.woff2 Sopensans-400.woff2 Sopensans-400.woff2 Sopensans-300.woff2 Sope	ame																			
tips.html collection.html guery-builderjs captchajs messages.js spinnerjs resolverjs adapterjs refreshjs authentication-popup.html messages.html captcha.html gallery.png mg03-br-0jpg 207 requests 1.1 MB transferred 207 requests 1.1 MB transferred 208 m	select-	bg.svg																		
collection.html query-builder,is captch.html galtery.png mg03-br-0,ipg 207 requests 1.1 MB transferred Console Sources Network Performance Memory Application Sec Console Sources Network Performance Memory Application Sec Sources Sources Network Performance Memory Application Sec Sources Sources Network Performance Memory Application Sec Sources Network Performance Memory Application Sec Sources Sources Network Performance Network Performanc	tips.ht	ml																		
guery-builder,is aptcha,is messages,is spinner,is resolver,is adapter,is refresh,is authentication-popup.html messages,html galtery,png mg03-br-0,jpg 207 requests 1.1 MB transferred Image: Sources in the second	collect	ion.htr	nl																	
captchajs messages,js spinner,js resolver,js adapter,js refresh,js authentication-popup.html messages.html captcha.html gallery.png mg03-br-0.jpg 207 requests 1.1 MB transferred COT requests 1.1 MB transferred	query-	builder	r.js																	
Inessages,js spinner,js resolver,js adapter,js refresh,js authentication-popup.html messages.html aptcha.html gallery,png mg03-br-0,jpg 207 requests 1.1 MB transferred Coroup by frame Preserve log Disable cache Offilter Hide data URLs XHR JS CSS Img Media Font Doc WS Manifest ame summit-watch.html e3ba05b55d9450782939b255da3e59ca.min.css styles.css logo.svg loader-1.gif 8502028c7c49cc5469f4a7565c494173.min,js Luma-tcons.woff2 fonts_e3ba05b55d9450782939b255da3e59ca.min.css print.min.css persams-400.woff2 opensams-400.woff2 opensams-400.woff2 opensams-400.woff2 opensams-600.woff2 gallery,png mg03-br-0,jpg	captch	a.js																		
spinnerjs resolverjs adapter js refreshjs authentication-popup.html messages.html captcha.html gallery.png mg03-br-0.jpg 207 requests 1.1 MB transferred Coroup by frame Preserve log ⊘ Disable cache Offi ter Hide data URLs (N KHR JS CSS Img Media Font Doc WS Manifest ame summit-watch.html e3ba05b55d9450782939b255da3e59ca.min.css styles.css logo.svg loader-1.gif 8502028c7c49cc5469f4a7565c494173.min.js Luma-tons.woff2 fonts_e3ba05b5d9450782939b255da3e59ca.min.css pint.min.css opensans-400.woff2 opensans-400.woff2 opensans-400.woff2 gallery.png mg03-br-0.jpg	messag	ges.js																		
resolver,is adapter,is refresh,is authentication-popup.html messages.html captcha.html gallery.png mg03-br-0,jpg 207 requests 1.1 MB transferred COT requests 1.1 MB transferred Styles-Link COT Property 2.1 MB transferred Styles-Cot Property 2.1 MB transferred 2.1 MB t	spinne	r.js																		
adapter,is refresh,is authentication-popup.html messages.html captcha.html gallery.png mg03-br-0jpg 207 requests 1.1 MB transferred Correct Console Sources Network Performance Memory Application Sec Correct Console Sources Network Performance Memory Application Sec Sources Network Performance	resolve	er.js																		
refresh,is authentication-popup.html messages.html captcha.html gallery.png mg03-br-0.jpg 207 requests 1.1 MB transferred Coroup by frame Performance Memory Application Sec Coroup by frame Preserve log Colsable cache Coff itter Hide data URLs Coroup by frame Preserve log Colsable cache Coff itter Hide data URLs Coroup by frame Preserve log Colsable cache Coff itter Hide data URLs Coroup by frame Preserve log Colsable cache Coff itter Hide data URLs Coroup by frame Preserve log Colsable cache Coff itter Hide data URLs Coroup by frame Preserve log Coroup by Manifest ame summit-watch.html e3ba05b55d9450782939b255da3e59ca.min.css styles.css logo.svg loader-1.gif 8502028c7c49cc5469f4a7565c494173.min.js Luma-Icons.woff2 poensans-700.woff2 opensans-700.woff2 opensans-700.woff2 opensans-700.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	adapte	er.js					(_			_	_							
authentication-popup.html messages.html captcha.html gallery.png mg03-br-0.jpg 207 requests 1.1 MB transferred Composition Sectors Composition Sectors Styles-Limin.css Styles-Limin.css Styles-Casecostors	refresh	n.js										_		_						
messages.html gallery.png mg03-br-0.jpg 207 requests 1.1 MB transferred C Elements Console Sources Network Performance Memory Application Sec C View: C O Hide data URLs View: C O View: C C] authen	nticatio	n-poj	pup.h	tml			F	2 F		- () [2 F	-						
captcha.html gallery.png mg03-br-0.jpg 207 requests 1.1 MB transferred C Elements Console Sources Network Performance Memory Application Sec C C C V View: T C View: T C View: T C View: T Hide data URLs XHR JS Summit-watch.html e3ba05b55d9450782939b255da3e59ca.min.css styles.css logo.svg loader-1.gif 8502028c7c49cc5469f4a7565c494173.min.js Luma-tcons.woff2 opensans-400.woff2 opensans-400.woff2 opensans-700.woff2 opensans-600.woff2 gallery.png gallery.png gallery.png gallery.png gallery.png	messag	ges.htr	nl							_ •	U		1	-						
gallery.png mg03-br-0.jpg 207 requests 1.1 MB transferred C Elements Console Sources Network Performance Memory Application Sec C C C View: C View: E C View: E C Support View: E C Support View: E C Support Support View: E C Support <] captch	a.html					L		_	_										
mg03-br-0.jpg 207 requests 1 1.1 MB transferred Image: Summit watch.html 9ba05b55d9450782939b255da3e59ca.min.css styles-Lmin.css styles.css loader-1.gif 850028c7c49cc5469f4a7565c494173.min.js Luma-Icons.woff2 popensans-700.woff2 opensans-700.woff2 opensans-700.woff2 opensans-600.woff2 opensans-600.wo] gallery	.png																		
207 requests 1.1 MB transferred	mg03-l	br-0.jpg	9																	
Elements Console Sources Network Performance Memory Application Sect Image: Sources Image: Sour		207	7 requ	Jests	T	1.1 M	1B tra	nsfer	red											
Elements Console Sources Network Performance Memory Application Sec Image: Sources Image: Sourc																				
New York View: Hide data URLs Hide data URLs WHR Jsummit-watch.html e3ba05b55d9450782939b255da3e59ca.min.css styles-Lmin.css styles.css logo.svg loader-1.gif 850208c7c49cc5469f4a7565c494173.min.js Luma-lcons.woff2 fonts_e3ba05b55d9450782939b255da3e59ca.min.css print.min.css opensans-400.woff2 opensans-00.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg		Ele	emen	ts	Со	nsole		Source	es	Netv	vork	Pe	erforn	nance	e Me	mory	Ар	plicat	ion	Sec
Iter Hide data URLs XHR JS CSS Ing Media Font Doc WS Manifest ame summit-watch.html Image: Strate St			7	Q	V	/iew:	:=	$\overline{\gamma_{n}}$		Group	by fra	me		Pres	erve log	Image: D	isable	cach	e	Of
ame summit-watch.html e3ba05b55d9450782939b255da3e59ca.min.css styles-l.min.css styles.css logo.svg loader-1.gif 8502028c7c49cc5469f4a7565c494173.min.js Luma-lcons.woff2 fonts_e3ba05b55d9450782939b255da3e59ca.min.css print.min.css opensans-400.woff2 opensans-700.woff2 opensans-300.woff2 opensans-300.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	ilter						Hide	data	URLs	All	XHR	JS	CSS	Img	Media	Font	Doc	WS	Mai	nifest
summit-watch.html e3ba05b55d9450782939b255da3e59ca.min.css styles-l.min.css styles.css logo.svg loader-1.gif 8502028c7c49cc5469f4a7565c494173.min.js Luma-lcons.woff2 fonts_e3ba05b55d9450782939b255da3e59ca.min.css print.min.css opensans-400.woff2 opensans-700.woff2 opensans-300.woff2 gallery.png mg03-br-0.jpg	ame																			
e 3ba05b55d9450782939b255da3e59ca.min.css styles-l.min.css styles.css logo.svg loader-1.gif 8502028c7c49cc5469f4a7565c494173.min.js Luma-lcons.woff2 fonts_e3ba05b55d9450782939b255da3e59ca.min.css print.min.css opensans-400.woff2 opensans-700.woff2 opensans-600.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	summi	t-watcl	h.htm	1																
styles-l.min.css styles.css logo.svg loader-1.gif 8502028c7c49cc5469f4a7565c494173.min.js Luma-lcons.woff2 fonts_e3ba05b55d9450782939b255da3e59ca.min.css print.min.css opensans-400.woff2 opensans-700.woff2 opensans-600.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	e3ba0	5b55d9	94507	8293	9b2	55da	3e59	ca.mi	n.css											
styles.css logo.svg loader-1.gif 8502028c7c49cc5469f4a7565c494173.min.js Luma-lcons.woff2 fonts_e3ba05b55d9450782939b255da3e59ca.min.css print.min.css opensans-400.woff2 opensans-700.woff2 opensans-300.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	styles-	l.min.c	ss	0295				cum												
logo.svg loader-1.gif 8502028c7c49cc5469f4a7565c494173.min.js Luma-lcons.woff2 fonts_e3ba05b55d9450782939b255da3e59ca.min.css print.min.css opensans-400.woff2 opensans-700.woff2 opensans-700.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	styles	css																		
loader-1.gif 8502028c7c49cc5469f4a7565c494173.min.js Luma-Icons.woff2 fonts_e3ba05b55d9450782939b255da3e59ca.min.css print.min.css opensans-400.woff2 opensans-700.woff2 opensans-300.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	logo.sv	/a																		
8502028c7c49c5469f4a7565c494173.min.js Luma-Icons.woff2 fonts_e3ba05b55d9450782939b255da3e59ca.min.css print.min.css opensans-400.woff2 opensans-700.woff2 opensans-300.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	loader-	-1.aif																		
Luma-lcons.woff2 fonts_e3ba05b55d9450782939b255da3e59ca.min.css print.min.css opensans-400.woff2 opensans-700.woff2 opensans-300.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	850202	28c7c4	9cc54	169F4	a75	65c4	9417	3.min	is											
fonts_e3ba05b55d9450782939b255da3e59ca.min.css print.min.css opensans-400.woff2 opensans-700.woff2 opensans-300.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	Luma-I	cons.w	off2						.,_											
print.min.css opensans-400.woff2 opensans-700.woff2 opensans-300.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	fonts (e3ba05	5b55c	9450)782	2939b	2550	la3e5	9ca.r	nin.css	;									
opensans-400.woff2 opensans-700.woff2 opensans-300.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	print.m	nin.css																		
opensans-700.woff2 opensans-300.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	opensa	ans-400).wof	f2																
opensans-300.woff2 opensans-600.woff2 gallery.png mg03-br-0.jpg	opensa	ans-700).wof	f2																
opensans-600.woff2 gallery.png mg03-br-0.jpg	opensa	ans-300).wof	f2									۸		— –					
] gallery.png] mg03-br-0.jpg	opensa	ans-600).wof	f2									А	F		:R				
mg03-br-0.jpg	gallerv	.png												•	-		•			
	ma03-l	br-0.ipd	3								C									
			-																	

16 requests | 565 KB transferred | Finish: 6.12 s | DOMContentLoaded: 4.99 s | Load: 5.11 s

CSS

O CSS:

Merge CSS Files		
[store view]	Yes (Recommended)	•
	'bin/magento setup:static-content:deploy' command m be run in the console when any of the settings in the fi been changed.	iust eld
Merge CSS Files in Admin Area [global]	No (Recommended)	• 🕜
Exclude URLs from CSS Merge [store view]		
	Each URL or part of the URL from new line	
Minify CSS Files [store view]	Yes (Recommended)	•
	'bin/magento setup:static-content:deploy' command m be run in the console when any of the settings in the fi been changed.	iust eld

Merging CSS Files works the same way as merging JS files.

Merge CSS Files in Admin Area - set *No* to forbid CSS files merging in the admin area. It won't affect your Goggle rating.

Exclude URLs from CSS Merge if needed.

Minifying CSS Files works the same way as minifying JS and HTML.

Defer Fonts Loading [store view]	Yes (Recommended) Fonts will be loaded later in a separate css file. 'bin/magent setup:static-content:deploy' command must be run in the	•
Defer Fonts for [store view]	Console when any of the settings in the field been changed Desktop Tablet	
Do Not Defer Fonts That Contain	Mobile Luma-Icons	
[store view]		_/
	Example: Luma-Icons	
Move Print CSS Files to Page Bottom [store view]	Yes (Recommended)	•

If you set **Defer Fonts Loading** to *Yes*, the browser will load the page with system fonts and replace them with your fonts when they are loaded. It makes store pages load faster.

The following command must be run in the console when any of the settings in the field been changed:

bin/magento setup:static-content:deploy

23/66

Defer Fonts for - choose the particular devices for which the fonts should be deferred.

You may exclude some fonts from deffering in the **Do Not Defer Fonts That Contain** option.

As in the JS tab, you can also **Move Print CSS Files to Page Bottom** to make your page render faster.

Flat Tables

2025/08/01 13:55

Flat Tables allow storing all the information about a category or a product. Such tables help to shorten the query in the database.

○ Flat Tables:	
Use Flat Catalog Category [global]	No (Recommended)
Use Flat Catalog Product [global]	No (Recommended)

Keep in mind that flat catalog usage is not recommended by Magento documentation. See more details here.

Server Push/Preload

A powerful feature of HTTP/2 represents the ability of the server to send multiple responses for a single client request. That is, in addition to the response to the original request, the server can push additional resources to the client, without the client having to request each one explicitly.

Server Push/Preload:

Enable Server Push/Preload [store view]	Basic Please note that in Chrome version up to 106 and other Chromium-based browsers will have Preloa functionality in their next releases. Server Push w work in Chrome version up to 105 and all other browsers as usual.	▼ (d ill
Asset Types to Server Push/Preload [store view]	Preloaded Images JS Files CSS Files Font Files	4
Exclude URLs from Server Push/Preload [store view]	example.com	1.

Each URL or part of the URL from new line

Please note that in Chrome version up to 106 and other Chromium-based browsers will have Preload functionality in their next releases. Server Push will work in Chrome version up to 105 and all other browsers as usual.

Before enabling the option, the server needs to be configured correspondingly for the proper work of the functionality. Please, follow the steps below to configure the server you're using.

Nginx Server

Please, add the following changes to the configuration file which is responsible for Magento:

- 1. Specify http2 flag for listen parameters
- 2. Add http2_push_preload on; to enable pushes (it should be under server scope)

3. Specify max pushes per request in http://www.specifymax_concurrent_pushes 50

```
server {
    listen 80;
    listen [::]:80;
    listen 443 ssl http2; <-- Add http2 flag
    listen [::]:443;
    http2_push_preload on; <-- Enable preload
    http2_max_concurrent_pushes 25; <-- Setup max pushes per request
    ...</pre>
```

Apache Server

1. First, you need to enable the HTTP/2 module.

sudo a2enmod http2

2. Then edit your Apache virtual host file. If you enabled HTTPS with Let's Encrypt, then the SSL virtual host is created as a separate file ending with **le-ssl.conf**.

sudo nano /etc/apache2/sites-enabled/your-site-le-ssl.conf

3. Put the following directive after the opening **<VirtualHost *:443>** tag.

Protocols h2 http/1.1

<u>a</u>masty

Last update: 2025/07/30 magento_2:google_page_speed_optimizer https://amasty.com/docs/doku.php?id=magento_2:google_page_speed_optimizer 13:37



4. Save and close the file. Then restart Apache for the changes to take effect.

sudo systemctl restart apache2

To make the feature work with **Varnish**, proceed to the Varnish configuration file and remove the following line from it:

unset resp.http.Link

Module Configuration

Now we can proceed to extension settings.

Enable Server Push/Preload - enable the **Basic** option to activate server push or Advanced to preload critical resources & assets.

Asset Types to Server Push - specify the files that will be used for the push.

Exclude URLs from Server Push - provide particular URLs to exclude from the push.

Server Push/Preload: Advanced

Please note that the advanced preload functionality is available only as a part of an active support or product subscription. You can find the **amasty/module-optimizer-subscription-functionality** package for installation in composer suggest.

27/66

Server Push/Preload:

Enable Server Push/Preload	Advanced		•	?		
[store view]	Please note that in Ch Preload functionality i all other browsers as i	rome version up to 106 and other C n their next releases. Server Push w usual.	Chromium-based bro vill work in Chrome v	owsers will have version up to 105 a	nd	
Preload URLs [store view]	Туре	Resource URL	Priority	Preload as	Enable CORS support (for Fonts Only)	Action
	Prel 🔻	static/version1737372995	Bro 💌	Ima 🔻	No 💌	1
	Preconnect		Bro 💌	Ima 🔻	No 🔻	1
	Add					

1. **Type**

- Preload
- Preconnect learn more here.
- DNS-prefetch learn more here.
- 2. Resource URL specify the URL of the asset to preload (Image, JS File, CSS File, Font File).

3. Priority

- Browser Priority (Recommended) refer to this documentation for more information.
- High
- Low

4. Preload as

- Image
- JS File
- CSS File
- Font File

5. Enable CORS support (for Fonts Only) - set to Yes to activate CORS support for fonts.

Other Settings

In this tab you can enable some additional settings to make your webstore perform better.

⊘ Other Settings:		
Sign Static Files [store view]	Yes (Recommended)	•
Asynchronous indexing [global]	Enable (Recommended)	•

Sign Static Files - activate the option to make the browser automatically update cached values once they are changed.

Asynchronous indexing - enable to distinguish traffic patterns on the database level to avoid conflicts between read and write operations. Order data is placed in a temporary storage and moved to the Order Management grid with no accidents.

Image Optimizer

Switch to this tab to configure the automatic image optimization using various flexible settings. The extension allows you to optimize JPEG, PNG and GIF images.

The correct functioning of the extension (Community and Enterprise editions) requires some special software to be installed: JPEG optimization tool / PNG optimization tool / GIF optimization tool / WEBp tool. If you are not sure how to install the necessary software on the server correctly, please contact your administrator or Amasty support team.

NOTE: Since the required third-party libraries cannot be installed on Magento Cloud, the image optimization functionality of the Google Page Speed Optimizer extension is not available for this edition. To enable image optimization on Magento Cloud, we recommend opting for the integration with SpeedSize[™].

SpeedSize Settings

In this tab, you can activate the integration with the SpeedSize[™] functionality for neuroscience media optimization. To learn more about SpeedSize[™], please check here.

Image Optimizer 2.0.0 by **amasty**

REQUEST NEW FEATURE

SpeedSize Settings

Public Key [global]

•••••

CDN [global]

Public Key - insert the key generated on the SpeedSize side.

CDN - specify CDN.

General Settings

Expand the **General Settings** tab to configure the main settings.

General Settings

Enable Multi-Process Generation [global]	No	?
Process Images per Request [global]	10	?
Image Resolutions Library [global]	GD2 🔻	?
	GD2	
	Imagick	
Optimize Images for Enabled Products Only [global]	Yes 🔻	
WebP Compression Quality [global]	75	?
AVIF Compression Quality [global]	30	?
AVIF Encoding Speed	6	?
	Allows to configure the speed of the encoding process, with higher values resulting in faster encoding but lower quality. Ranges from 0 (slow, smaller file) to 10 (fast, larger file).	

Enable Multi-Process Generation - select Yes to speed up images optimization.

Multi-Process Optimization significantly boosts the image optimization speed. But it generates extra load to the server as well. We recommend first to run the test image optimization process with the 'multi-process' *ON* and monitor your server performance.

The 'Multi-Process Generation' feature needs the php extension '**pcntl**' to be installed on the server. If you enable the feature and no performance boost happens, please ask your hoster/system administrator to check if the 'pcntl' extension installed.

Number of Parallels Processes - specify the number of processes to be performed at once. The more parallel processes are set, the faster the image optimization process is, as well as the higher is the server load.

Image Resolutions Library - Choose the library to use for creating image resolutions during Smart

Optimization. The choice of library affects the speed of image optimization. GD2 is the default library, while using the Imagick library can speed up the optimization process, especially for large product catalogs.

For the Imagick option to work, the corresponding library must be installed. You can download it using the following link.

Process Images per Request - enter the number of images you want to be shown instantly, per one request.

You may also optimize images in bulk using a console command (here **4** is the number of threads):

bin/magento amasty:optimizer:optimize -j 4

Moreover, you can specify the particular pattern (**i** is the id of the pattern):

bin/magento amasty:optimizer:optimize -j 4 -i 2

To optimize a specific image, run the following command (-i is the ID of the pattern, wysiwyg/image.jpg is the image path):

bin/magento amasty:optimizer:optimize-img -i 2 wysiwyg/image.jpg

Please note that the last command is available only as a part of an active support or product subscription. You can find the **amasty/module-optimizer-subscription-functionality** package for installation in composer suggest.

Optimize Images for Enabled Products Only - enable this option if you want to create smaller copies for the images of enabled products only. It helps to save extra space on your server and speed up optimization.

WebP Compression Quality - set the required compression quality for WebP images (from 50 to 99).

AVIF Compression Quality - specify the desired compression quality for AVIF images. The quality level can be adjusted on a scale from 0 (worst quality, smaller file) to 100 (best quality, larger file).

AVIF Encoding Speed - configure the speed of the AVIF encoding process. The speed can be adjusted on a scale from 0 (slow, smaller file) to 10 (fast, larger file).

To learn more, click the following link.

User Agent

Expand the **User Agent** tab to adjust settings for image loading according to the customer's browser user agent.

Please note that the User Agent functionality works only with Smart Optimization strategy.

User Agent

Use User Agent for Loading Images [store view]	Yes	· ?
	Please note that the User Agent functionality works only with Smart Optimization strategy.	
Ignore Images that Contain [store view]	<img <br="" class="testImage lazyOwl"/> src="http://site.com/1.png"/> 	
	Example: <img <="" class="testImage lazvOwl" td=""/> <td>11</td>	11
	src="http://site.com/1.png"/> OR will be excluded.</img 	

Set the **Use User Agent for Loading Images** to **Yes** to enable fetching appropriate image sizes based on the user agent data, obtained from the browser.

In the **Ignore Images that Contain** you can specify certain images to be excluded from the loading based on the user agent info. Such images will be delivered on the users' devices in the original size.

Use EOL separator for your images ignore list. Several records should be written on a new line.

Replace Images

Choose the strategy of image optimization. Here you can implement the options selected in the New Pattern for Image Optimization settings section.

Replace Images

Image Replace Strategy [store view]	Smart Optimization	•	?
	None		
Ignore Images that Contain	Smart Optimization		
[store view]	SpeedSize		
		4	
	Specify each record from a new line.		
Support Third-party Image Attributes [store view]	<img original-image.png"="" src="placeholder-image.png
data-src="/>		?
		- 1	
	Specify each attribute from a new line or le empty if it shouldn't be used.	eave	

Image Replace Strategy - select the suitable strategy for image optimization:

* Choose None if you don't need to optimize images;

* Choose the **Smart Optimization** strategy to automatically replace all existing images with lighter alternatives in either **AVIF** or **WebP** format. When this strategy is enabled, images will be replaced with AVIF format versions. If the user's browser doesn't support the AVIF format, images will be replaced with WebP format alternatives (WebP and Image Resolutions strategy). You can access a list of browsers that support the AVIF format here;

AVIF is a modern image format based on the AV1 video format, known for its superior compression compared to WebP, JPEG, PNG, and GIF. It is designed to replace them as the preferred image format. Please be informed that the **Smart Optimization** strategy (AVIF format) is available only as part of an active product subscription or support subscription for Pro and Premium extension versions.

* Choose **SpeedSize** to optimize all existing images with the SpeedSize[™] functionality. Please note that this strategy will work only if the correct CDN and Public Key in the SpeedSize Settings section are specified.

SpeedSize - a solution based on neurobiological optimization of multimedia that allows you to maintain high quality and speed of loading images.

Video support: if SpeedSize is enabled (CDN and Public Key are filled out), the original video links on the pages will be replaced with the SpeedSize links without additional configuration regardless of whether the SpeedSize strategy is selected or not.



Ignore Images that Contain - specify particular images that should be skipped.

Use EOL separator for your images ignore list. Several records should be written on a new line.

Support Third-party Image Attributes - you can replace images generated by third-party lazy load extensions by specifying the required attribute. Example: In this case 'data-src' - it's a third-party attribute, so you need to prescribe it in the field if you want images with this attribute to be replaced with WebP format.

Additionally, you can adjust the optimization for particular store pages separately. The settings are the same as above.

🔗 Home Page			
Enable Custom Image Replace [store view]	Yes	•	
Custom Image Replace Strategy [store view]	Smart Optimization	•	
Ignore Images that Contain [store view]	SpeedSize		
Support Third-party Image Attributes [store view]	Specify each record from a new line.	//	2
	empty if it shouldn't be used.		
⊘ Category Pages			
⊘ Product Pages			
⊙ CMS Pages			

Automatic Image Optimization

In this tab, you can allow automatic image optimization.

Automatic Image Optimization

 Automatically Optimize Newly Uploaded Images [store view]
 Yes (Recommended)
 •

 Add folders to automatic optimization in the pattern settings.

 Automatically Optimize Images Upgrade Your Plan
 Immediately after Uploading
 •

 The functionality is available in the Premium tariff of the Google Page Speed Optimizer module. To upgrade and obtain functionality please follow the link.

Automatically Optimize Newly Uploaded Images - set to *Yes* to automatically optimize all newly uploaded images in folders where the *Automatically Optimize Newly Uploaded Images* setting in the pattern settings is also set to *Yes*.

Please note that when you install the extension, you have to optimize all previously uploaded images. It can be done from the **Content** → **Image Optimizer** → **Image Folder Optimization Settings page**. But if you upload new images after the optimization (e.g. a new product has been created), use this option to optimize the newly uploaded images on the fly before saving it. Upgrade to **Premium** and subscribe to access automatic optimization for existing images in folders

and delayed optimization by cron based on a schedule.

Delete Previously Generated & Resized Images



Clear Images Folder - the Clear WebP Images Folder / Clear Avif Images Folder/ Clear Mobile Images

<u>a</u>masty

Folder / Clear Tablet Images Folder buttons delete the optimized files from the **pub / media / amasty / folder_name** folders, depending on which button is pressed. It is used if the previously selected optimization settings were inconvenient to use and there is a need to re-generate the queue and re-optimize the images.

Lazy Load

This extension is used to load images only when they become visible, which makes pages load faster. Proceed to the **Lazy load** tab to enable the fucntionality.



To deliver the media correctly, the **srcset** attribute will be deleted from Lazy Load images.

General

Enable Module - set to Yes to activate lazy load functionality.

User Agent

If you have **Image Optimizer** installed, you will be able to configure the user agent for lazy load functionality. Expand the **User Agent** tab to adjust settings for image loading according to the customer's browser user agent.

Please note that the User Agent functionality does not work when SpeedSize is enabled.

User Agent



Set the **Use User Agent for Loading Images** to **Yes** to enable fetching appropriate image sizes based on the user agent data, obtained from the browser.

In the **Ignore Images that Contain** you can specify certain images to be excluded from the loading based on the user agent info. Such images will be delivered on the users' devices in the original size.

Lazy Load is fully compatible with the Full Page Cache Warmer extension. If you have both modules installed, you can warm pages according to the user agents.

 \bigcirc

Lazy Load Compatibility Settings

User Agents [global] WebP Support	?
No WebP Support	
Images	
Resolution	
[global] Tablet	
Mobile	

Lazy Load

Lazy Load

Use Lazy Loading Images			
[store view]	Yes (Recommended)	•	
Lazy Load Script			
[store view]	jQuery Lazy Script	•	
Duala ad Issa as a	I		
[store view]	Yes (Recommended)	•	?
Enable SpeedSize [store view]	No	•	
Desktop Preload Images Number	10		
Tablet Preload Images Number	7		
[store view]			
Mobile Preload Images Number	4		
[store view]			
Ignore Preload Images that Contain	product-image-phot		
[store view]	product mage proc		
		/	
Ignore Images that Contain	lazyOwl		
[store view]			
		11	
	Specify each record from a new line.		
Add Aspect Ratio Styles to Image Tags	Yes	•	?
Excel County			
Add to Images that Contain	aspect-ratio		
[store view]			
		4	
	"Aspect-ratio" will be added to images that contain <ing aspect-ratio.png"="" class="testIr</td><td>nage</td><td></td></tr><tr><td></td><td>src=" http:="" site.com=""></ing> OR <img< td=""><td></td><td></td></img<>		

Use Lazy Loading Images - set to *Yes* to make off-screen images load only when a customer scrolls to them. It prevents pages full of images from lagging.

Lazy Load Script - you can choose between 4 lazy load scripts: *jQuery Lazy Script, Native JS Lazy Script, Vanilla Lazy Script,* and *Lozad Lazy Script.* We recommend trying all of them to choose one that fits your system best.

Please note, that for the correct work of Lazy Load functionality on Hyva Theme, the option Native

Last update: 2025/07/30 magento_2:google_page_speed_optimizer https://amasty.com/docs/doku.php?id=magento_2:google_page_speed_optimizer 13:37

JavaScript Lazy Script must be chosen.

If you have the Pro or Premium version of the extension along with an <u>active product subscription or</u> <u>support subscription</u>, you can utilize the lazy load functionality for images with the **<source>** tag within **<picture>** and images with the **<srcset>** tag within ****. However, it's important to note that optimization and lazy loading for images with the **<source>** tag in **<picture>** will only work if one of the following scripts is selected: Vanilla, jQuery, or Lozad Lazy script.

You can decide whether to request the preload of Magento 2 key images or load them from scratch when they happen on screen. Decide how many images to load by setting **Preload Images Number**.

Enable SpeedSize - set Yes if you want to enable SpeedSize for lazy load (for all pages at once).

Desktop Preload Images Number - set the number of images to be pre-loaded on the page from the start. You can also assign a specific number of images for tablet devices with the **Tablet Preload Images Number** and for phones with the **Mobile Preload Images Number** fields.

Ignore Preload Images that Contain - specify the images (their classes) that should be excluded from preloading.

Ignore Images That Contain - specify the images that should not be optimized.

Use EOL separator for your images ignore list. Several records should be written on a new line. For better compatibility with other extensions, the **img** tag which includes empty **src** will be ignored.

Add Aspect Ratio Styles to Image Tags - if enabled, the aspect-ratio styles will be added to lazy loaded images based on their dimensions. It helps you to display images correctly even when only image width or height is specified.

Add to Images that Contain - specify particular images or the pattern to which the aspect ratio style should be added.

How it works: if you specify **myimage** in this field, "Aspect-ratio" style will be added to images that contain **myimage** in its path or class. For example:

```
<img class="testImage myimage" src="http://site.com/1.png"/>
```

OR

Please note that when the Use User Agent for Loading Images setting is set to **No** and the **Preload Images** setting is set to **Yes**, the following features will be available:

<u>a</u>masty

Preload Images Strategy [store view]	Smart Optimization	•
Preload Images Count [store view]	7	

To learn more about Image Strategies, navigate here.

All other tabs of this block have the same structure, but you can specify the settings for particular site pages. Thus, you can select the most suitable type of optimization for each store page.

⊘ Home Page

Enable Custom Lazy Load [store view]	Yes	
Use Lazy Loading Images [store view]	Yes (Recommended)	
Lazy Load Script [store view]	jQuery Lazy Script	
Preload Images [store view]	Yes (Recommended)	0
Desktop Preload Images Number [store view]	1	
Tablet Preload Images Number [store view]	1	
Mobile Preload Images Number [store view]	1	
Ignore Preload Images that Contain [store view]		
		4
Ignore Images that Contain [store view]		
		4
	Specify each record from a new line.	
Ocategory Pages		
O Product Pages		
⊙ CMS Pages		

Save the configuration.

AMP

Pro and Premium extension versions include AMP extension. The collaboration of these modules lets you create accelerated mobile pages and use Webp image format for them, which significantly

improves mobile score.

To configure the AMP component, proceed to **Stores** \rightarrow **Configuration** \rightarrow **Amasty Extensions** \rightarrow **AMP**.

Expand the General tab.

General

Enable AMP for Product Pages [store view]	Yes	•
Enable AMP for Category Pages [store view]	Yes	•
Enable AMP for Home Page [store view]	Yes	•
Enable AMP for CMS Pages [store view]	Yes	•
Open Accelerated Mobile Pages [store view]	From AMP Google results only	•
	From AMP Google results only Always	

Choose the types of pages for which the AMP versions will be generated.

Open Accelerated Mobile Pages - if *Always* option is chosen, users on mobile devices will be invariably redirected to AMP version of the page. If you select *From AMP Google results only*, AMPs will be available only via links on Google search results pages (in that case the user will stay on AMPs until he/she reaches non-AMP page).

You can also enable AMP for CMS pages and configure design options. Check this user guide to see the details.

Image Folder Optimization Settings

These settings will allow to precisely configure the optimization pattern for folders.

Navigate to **Content** → **Image Optimizer** → **Image Folder Optimization Settings**. To create new set of rules, please click on the **Add New Pattern** button.

Image Folder Optimization Settings					
		Run Optimization	Add New Pat	tern	
Actions 💌 2	2 records found	Filters Oefat 20 r	ult View 👻 🏠 Colo	umns 👻	
D I	Folders	Title	Status	Action	
1	wysiwyg	wysiwygFolders	Enabled	Edit	
2	wysiwyg/mens	wysiwyg/mensFolder	Enabled	Edit	

All configured patterns will be displayed in one handy grid. You can manage them any time you need just in a few clicks.

Run Optimization - click this button to optimize selected folders images.

To run optimization manually, execute the following command:

php bin/magento amasty:optimizer:optimize

Please mind that running optimization will make an additional load to the server.

New Pattern for Image Optimization settings

As you click the Add New Pattern button, a new Image Optimization Settings page will expand.

Image Optimization Settings

Please note, that it is possible to launch queue generation and optimization with the following CLI command:

php bin/magento amasty:optimizer:optimize		්) Copy
---	--	---------

You may also optimize images in bulk using the following command (here 4 is the number of threads):

bin/magento amasty:optimizer:optimize -j 4

🖒 Copy

Moreover, you can specify the particular pattern (-i is the ID of the pattern):

bin/magento amasty:optimizer:optimize -j 4 -i 2

🖒 Copy

To optimize a specific image, run the following command (-i is the ID of the pattern, wysiwyg/image.jpg is the image path):

```
bin/magento amasty:optimizer:optimize-img -i 2 v
```

්ා Copy

To optimize a specific image instead of an entire folder, according to a pattern settings, run the following command (-i is the ID of the pattern, wysiwyg/image.jpg is the image path):

bin/magento amasty:optimizer:optimize-img -i 2 wysiwyg/image.jpg

With this command, you can also optimize multiple images at once by specifying their spaceseparated paths:

```
bin/magento amasty:optimizer:optimize-img -i 2 wysiwyg/image.jpg
wysiwyg/image2.jpg wysiwyg/image3.jpg
```

Please note that this command is available only as a part of an active support or product subscription. You can find the **amasty/module-optimizer-subscription-functionality** package for installation in composer suggest.

<u>a</u>masty

Enabled	Yes Yes
Title *	wysiwygFolders
Folders for Optimization *	wysiwyg ×
	Q
	😔 🗌 amasty
	Catalog
	custom_options
	customer
	customer_address
	🔆 🗌 downloadable
	· import
	sitemap
	Son theme
	theme_customization
	🔆 🗹 wysiwyg
	Done

Set the **Enabled** switch to **Yes** to allow this set of configurations.

Title - enter the title of the configuration set to easily distinguish numerous configs in the grid.

Folders for Optimization - click on the folder name to include it in the list of folders that will be optimized.

NEW: Magento 2 Google Page Speed Optimizer extension is compatible with **AWS Remote Storage** (**AWS S3**). That allows optimizing images from folders that are located in AWS Remote Storage. The names of the mentioned folders must be chosen in the *Folders for Optimization* field. After this, the images will be optimized according to the configuration of the specific pattern.

Thanks to the compatibility between Google Page Speed Optimizer Pro and Amasty's Blog Pro, you can include the folder (**amasty/blog**) containing your blog images during pattern configuration and optimize them according to the defined pattern settings — provided both modules are installed.

Au	tomatically Optimize Newly Uploaded Images in Folder	Yes ? Enabling the setting will automatically optimize newly uploaded images.
8	Automatically Optimize Existing Images in Folder Subscribe to Unlock	No The functionality is available as a part of an active product subscription or support subscription. To upgrade and obtain functionality, please follow the link. Then you can find the 'amasty/module-optimizer-premium- subscription-functionality' package for installation in composer suggest.
	Create Image for Mobile	Yes ?
	Create Image for Tablet	Yes Yes The extension will create a copy of every image in suitable for mobile or tablet size. Please make sure that you have enough disc space on your server.
	Resize Algorithm	Resize 🔹
	Create AVIF Copy	Cavif When AVIF format is used the extension will create a copy for every image in the AVIF format. Please make sure that you have PHP version 8.1 or higher and you have enough disc space on your server.
	Create Webp Copy	Cwebp When WebP format is used the extension will create a copy for every image in the WebP format. Please make sure that you have enough disc space on your server.
	Create Image Dump	Yes ?

Automatically Optimize Newly Uploaded Images in Folder - set to *Yes* to automatically optimize all images newly uploaded in the folders selected for optimization.

For correct work, please make sure that the **Automatically Optimize Newly Uploaded Images** setting is enabled in the general configuration (**Image Optimizer** \rightarrow the **Automatic Image Optimization** tab).

Upgrade to **Premium** and subscribe to unlock automatic optimization for existing images in folders and the ability to log image optimization after each cron job run.

Create Image for Mobile/Tablet - set the corresponding switch to **Yes** to automatically create appropriately sized images for various displays. All newly created copies will be saved on your website server, so check in advance if you have enough disc space.

Resize Algorithm - select the algorithm according to which the images will be compressed:

- **Crop** cuts pictures by size according to the device used, but not proportional to the size of the picture itself. While the height of the image would be the same, in the result the cropped image will look like the part of the original picture.
- **Resize** compresses the images in proportion to the picture itself according to the size of the field viewed. Simply it will be the same picture, but smaller in pixels.

Create AVIF Copy - select Cavif to generate an AVIF format copy for every image.

Please make sure that you have PHP version 8.1 or higher and you have enough disc space on your server.

Create Webp Copy - choose Cwebp to create a copy for every image in the WebP format.

Please make sure that you have enough disc space on your server. It is also necessary to install the Webp tool package on your server using the command

sudo apt-get install webp

For a specific setup with versioning, use this guide.

Create Image Dump - set to Yes to to store original images in the

pub/media/amasty/amoptimizer_dump folder. It allows collect all the original files in one place and easily update them in case the settings are changed.

JPEG Optimization Tool	Jpegoptim 80% quality	•
	New Luna Yoga Collection Get fit and look fab in new seasonal styles	
PNG Optimization Tool	Optipng	•
GIF Optimization Tool	Gifsicle	•

JPEG optimization tool - set the way to optimize JPEG images:

- Do not Optimize select this option to forbid optimization;
- jpegoptim 100% quality / jpegoptim 90% quality / jpegoptim 80% quality choose the quality level you want to get after compression. See the examples of optimized images right below the setting.

PNG optimization tool - select optipng to optimize PNG images.

GIF optimization tool - choose gifscale to compress GIF files.

Click on the **Save** button to preserve the settings.

PREMIUM: Advanced Features

The Premium version includes some unique features to improve store speed.

Premium JS Options

Since JS merging creates one huge file, for particular stores it may affect some pages on specific devices. The Premium version provides more flexibility - you can adjust merging for any device to achieve maximal effect.

To adjust the option, navigate to **Google Page Speed Optimizer** \rightarrow **JavaScript** and enable the *Merge JavaScript Files* feature.

 (\land)

JavaScript:

8

Please note that Magento Production mode should be set instead of Developer or Default to get the best performance of the settings below.

Amasty JS Optimization [store view]	Disabled	•	0
Merge JavaScript Files [global]	Yes 'bin/magento setup:static-content:deploy command must be run in the console wh any of the settings in the field been chan	y' hen nged.	
Disable JavaScript Merging for [store view]	Desktop Tablet Mobile		
Enable JavaScript Bundling [global]	No 'bin/magento setup:static-content:deploy command must be run in the console wh	▼ y' hen	

Please, note that the feature behaves differently for different stores and needs testing. Try to disable merging for particular devices on your testing environment to find the combination that suits your website best.

Automatic Optimization for Existing Images in Folders

Please note that this functionality is available only as a part of an active support or product subscription. You can find the **amasty/module-optimizer-premium-subscription-functionality** package for installation in composer suggest.

In addition to optimizing newly uploaded images automatically, the **Premium** extension version also enables automatic optimization of existing images in folders. To configure this:

1. First, go to Stores \rightarrow Configuration \rightarrow Google Page Speed Optimizer (Premium) \rightarrow Image Optimizer \rightarrow Automatic Image Optimization and enable the Automatically Optimize Existing Images setting:

tomatic Image Optimization		
Automatically Optimize Newly Uploaded Images [store view]	Yes (Recommended) Add folders to automatic optimization in the pattern settings.	?
Automatically Optimize Existing Images [store view]	Yes Add folders to automatic optimization in the pattern settings.	?

2. Then, navigate to **Content** \rightarrow **Image Optimizer** \rightarrow **Image Folder Optimization Settings** and activate the **Automatically Optimize Existing Images in Folder** feature:

Last update: 2025/07/30 13:37

Edit Image Folder Sett	ings	← Back	Delete	Save an	d Continue	Edit
Enabled Title *	Yes Product Images	Pattern				
Folders for Optimization * Automatically Optimize Newly Uploaded Images in Folder	amasty × ama	asty/amoptimizer_ will automatically o	dump × am	asty/amoptmol	s.	
Automatically Optimize Existing Images in Folder	Yes Enabling the setting v	will automatically o	optimize existing	; images.		
Create Image for Mobile	🚺 Yes 🕐					
Create Image for Tablet	Yes ? The extension will cre Please make sure tha	eate a copy of ever at you have enougl	y image in suita h disc space on y	ble for mobile o your server.	or tablet size.	
Resize Algorithm	Resize				•	?
Create AVIF Copy	Cavif When AVIF format is Please make sure tha on your server.	used the extension at you have PHP ve	n will create a co rsion 8.1 or high	py for every im her and you hav	age in the AVIF f e enough disc s	ormat. pace
Create Webp Copy	Cwebp When WebP format is WebP format. Please	s used the extensi make sure that yo	on will create a c ou have enough (copy for every ir disc space on y	mage in the our server.	0
Create Image Dump	🌔 Yes 🕐					

Existing images in selected folders will be optimized by cron on a scheduled basis. See the next user guide section for more details.

The number of existing images that can be optimized in a single cron run is controlled by the **Process Images per Request** setting in the general configuration (Image Optimizer \rightarrow the General Settings tab).

Image Optimization by Cron

The Pro version has an option for automatic image optimization - all new images are optimized as soon as they are uploaded. But for some stores it is necessary to delay this process, e.g. run optimization at night, when the number of visitors is minimal. The Premium version offers Cron image compression so that the optimization could suit your personal working flow.

To set up an optimization schedule, proceed to **Image Optimizer** \rightarrow **Automatic Image Optimization** and choose the required option in the *Automatically Optimize Images* dropdown.

Automatically Optimize Images [global]	By Cron Immediately after Uploading By Cron
Image Optimization Schedule [global]	0 */12 * * * Enter cron expression * * * * * + Day of the Week (range: 0-6, 1 standing for Monday) + Month of the Year (range: 1-12) + Day of the Month (range: 1-31) + Hour (range: 0-23) + Minute (range: 0-59) Example: 0 */12 * * * - every twelve hours Read more about cron expressions - here

Please, mind that when you switch the Cron optimization to the **Immediately after Uploading** mechanism, some images may still remain in the queue. To optimize images left in the queue, run the following command:

bin/magento amasty:optimizer:optimize

Image Optimization Log

The **Enable Image Optimization Log** feature exclusively available in the Premium extension version and located in the **General Settings** tab of the **Image Optimizer** general configuration, helps track the image optimization process in the background, ensuring all operations are properly documented.

Please note that the feature is available only as a part of an active support or product subscription. You can find the **amasty/module-optimizer-premium-subscription-functionality** package for

installation in composer suggest.

General Settings

Enable Multi-Process Generation [global]	No	?
Process Images per Request [global]	10	?
Image Resolutions Library [global]	GD2 For the Imagick option to work, the corresponding library must be installed. You can download it from the following	0
	link.	
Optimize Images for Enabled Products Only [global]	Yes	
WebP Compression Quality [global]	75	?
AVIF Compression Quality [global]	30	?
AVIF Encoding Speed	6	?
- <u>-</u>	Allows to configure the speed of the encoding process, with higher values resulting in faster encoding but lower quality. Ranges from 0 (slow, smaller file) to 10 (fast, larger file).	
Enable Image Optimization Log [global]	Yes 🔹	
	You can find the log file for automatic image optimization in the var/log folder, e.g. var/log/amoptimization_cron_%timestamp-Y-M-D%.log. This log is utilized to track the process of optimizing images in the background, ensuring all operations are properly documented.	

Enable Image Optimization Log – If set to **Yes**, a log file containing details of automatic image optimization (e.g., 'var/log/amoptimization_cron_%timestamp-Y-M-D%.log') will be created after each cron run. You can find these files in the 'var/log' folder.

The log file includes the following details:

- File (image) name
- Error message (if any)
- Time taken for generation

• Status (success or failure)

Log file example (click to zoom):

3 [2024-12-18T16:09:45.931959+00:00] 4 [2024-12-18T16:09:47.424862+00:00]	<pre>amoptimizer_logger.INF0: File amoptimizer logger.INF0: File</pre>	"wysiwyg/amasty/rma/approved_rma.jpg" has been successfully processed. [] [] "wysiwyg/amasty/rma/need details.ong" has been successfully processed. [] []
5 [2024-12-18T16:09:48.652781+00:00]	amoptimizer logger.INFO: File	"wysiwyg/amasty/rma/received admin.png" has been successfully processed. [] []
6 [2024-12-18T16:09:49.345104+00:00]	amoptimizer logger.INFO: File	"wysiwyg/amasty/rma/reject admin.png" has been successfully processed. [] []
7 [2024-12-18T16:09:50.819728+00:00]	amoptimizer logger.INFO: File	"wysiwyg/amasty/rma/resolved admin.png" has been successfully processed. [] []
8 [2024-12-18T16:09:51.559476+00:00]	amoptimizer logger.INFO: File	"wyslwyg/amasty/rma/resolved rated.png" has been successfully processed. [] []
9 [2024-12-18T16:09:53.021059+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/collection/collection-eco.jpg" has been successfully processed. [] []
10 [2024-12-18T16:09:54.620074+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/collection/collection-performance.jpg" has been successfully processed. [] []
11 [2024-12-18T16:09:54.722340+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/gear/gear-category-bags.jpg" has been successfully processed. [] []
[2 [2024-12-18T16:09:54.816080+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/gear/gear-category-equipment.jpg" has been successfully processed. [] []
<pre>I3 [2024-12-18T16:09:54.902320+00:00]</pre>	amoptimizer_logger.INFO: File	"wysiwyg/gear/gear-category-watches.jpg" has been successfully processed. [] []
4 [2024-12-18T16:09:55.925010+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/gear/gear-equipment.jpg" has been successfully processed. [] []
L5 [2024-12-18T16:09:56.597977+00:00]	amoptimizer_logger.INF0: File	"wysiwyg/gear/gear-fitnes.jpg" has been successfully processed. [] []
16 [2024-12-18T16:09:57.744771+00:00]	<pre>amoptimizer_logger.INF0: File</pre>	"wysiwyg/gear/gear-main.jpg" has been successfully processed. [] []
17 [2024-12-18T16:09:59.026626+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/giftcards/giftcard-mailed.jpg" has been successfully processed. [] []
<pre>18 [2024-12-18T16:10:00.338906+00:00]</pre>	amoptimizer_logger.INF0: File	"wysiwyg/giftcards/giftcard-virtual.png" has been successfully processed. [] []
19 [2024-12-18T16:10:01.738230+00:00]	amoptimizer_logger.INF0: File	"wysiwyg/giftcards/giftcard.jpg" has been successfully processed. [] []
0 [2024-12-18T16:10:03.319165+00:00]	amoptimizer_logger.INF0: File	"wysiwyg/home/home-eco.jpg" has been successfully processed. [] []
21 [2024-12-18T16:10:04.218136+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/home/home-erin.jpg" has been successfully processed. [] []
22 [2024-12-18T16:10:05.445887+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/home/home-main.jpg" has been successfully processed. [] []
23 [2024-12-18T16:10:05.719687+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/home/home-pants.jpg" has been successfully processed. [] []
24 [2024-12-18T16:10:06.808039+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/home/home-performance.jpg" has been successfully processed. [] []
25 [2024-12-18T16:10:06.856317+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/home/home-t-shirts.png" has been successfully processed. [] []
26 [2024-12-18T16:10:08.635458+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/logo.jpg" has been successfully processed. [] []
27 [2024-12-18T16:10:08.736909+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/mens/mens-category-hoodies.jpg" has been successfully processed. [] []
28 [2024-12-18T16:10:08.838702+00:00]	amoptimizer_logger.INFO: File	"wyslwyg/mens/mens-category-shorts.jpg" has been successfully processed. [] []
9 [2024-12-18116:10:08.924070+00:00]	amoptimizer_logger.INFO: File	"wyslwyg/mens/mens-category-tees.jpg" has been successfully processed. [] []
0 [2024-12-18T16:10:09.993364+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/mens/mens-main.jpg" has been successfully processed. [] []
S1 [2024-12-18/16:10:11.082244+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/mens/mens-pants.jpg" has been successfully processed. [j []
2 [2024-12-18110:10:11.108936+00:00]	amoptimizer_logger.INFO: File	"wysiwyg/mens/mens-t-snirts.png" processing failed. Error: "Test error". [] []
[2024-12-18110:10:11.109111+00:00]	amoptimizer_logger.INFO: Proce	ess ratient [j []
14 [2024-12-18/10:10:11.109222+00:00]	amoptimizer_togger.inFO: Exect	arrow true: 52:303210000122 [] []

AJAX Shopping Cart

To provide customers with the extra fast and handy shopping flow, the Premium version includes AJAX Shopping Cart extension. If enabled, shoppers are able to add and manage products in a shopping cart without redirects and page reloads.

Navigate to Stores → Configuration → Amasty Extensions → AJAX Shopping Cart.

General

Enable Ajax Shopping Cart [store view]	Yes 💌
Enable Flying Image Effect [store view]	Yes Set to 'Yes' for a nice-looking visual effect. The product image 'flies' to the 'Top Cart' block in the top menu.
Show the Product Qty Added to the Cart on the Product Listing [store view]	Yes
Add Products to Compare with AJAX [store view]	No
Add Products to Wish List with AJAX [store view]	No
Show a Mini Cart once Item is Added [store view]	No
	Set 'Yes' to Show a Mini Cart instead of Confirmation Pop- up once an item is added to the cart.

Enable or disable the extension in **Enable Ajax Shopping Cart**.

Enable flying image effect: With this setting, you can activate and deactivate nice looking visual effect.

You can get an idea of how the effect works here.

Show the Product Qty Added to the Cart on the Product Listing: Enable the option to highlight the products that have already been added to the shopping cart. With mouse hover over the product, the number of items added to the cart appears.

Add Products to Compare with AJAX: Enable the option to let customers add products to Compare without page reloading. This will fasten the process and make it more convenient.

Add Products to Wish List with AJAX: Set the option to Yes to eliminate page reloading when customers add products to Wish List.

Show a Mini Cart once Item is Added: Enable the option to Show a Mini Cart instead of a Confirmation Pop-up after an item is added to the cart.

The extension includes a wide range of cart customization settings. See this user guide to configure them properly.

<u>a</u>masty

General settings

Infinite Scroll

You can also find this functionality in the Premium version only. Infinite scroll lets your visitors browse the catalogs without page reloads, as the content is loaded and displayed as soon as a customer reaches the end of the catalog page.

To get access to the infinite scroll settings please go to Stores \rightarrow Configuration \rightarrow Amasty Extensions \rightarrow Infinite Scroll



Loading Type - choose from the available loading styles.

See the full list and each type specifics below:

- Select **None module is disabled** if you want to deactivate the functionality of the Infinite Scroll extension.
- Select **Automatic on page scroll** to activate the autoloading of the previous and next pages. The next and previous pages will be displayed as soon as you reach the end of the current page. The AJAX plugin provides the progressive enhancement of the next and previous pages to the existing page without a complete page refresh.
- If you select **Button on button click** you'll have the next and previous pages loaded automatically but they will be displayed only after you click the button.
- In case you choose the **Combined automatic + button**, the first 'X number' of pages will be loaded automatically. After X pages, the user will have to load pages manually by pressing the button.
- For **Combined button + automatic**, the first 'Y number' of pages will be loaded automatically. Then, to load the next page, the user will need to click a button once. After that, all other pages will be loaded automatically.

Number of Automatically Loaded Pages - Enter the number of pages a user needs to scroll before 'Auto mode' applies. This setting is only available for *Combined* loading types: *Combined - automatic* + *button* and *Combined - button* + *automatic*.

What is the difference between **automatic + button** vs **button + automatic**?

Let's say, you set the number of automatically loaded pages as 4.

In the case of the **Combined - automatic + button** scenario, a user will see the first 4 pages similarly to a long one-pager. Then, to load the 5th, 6th, 7th, etc. pages, the user will need to press a '*Load More*' button every time to load the page.

However, with the **Combined - button + automatic** user will be asked to press a '*Load More*' button only once - between the 4th and 5th page.

Display Page Numbers - Choose '*Yes*' to display the pages separator with numbers. While scrolling down the list of products you will see the line separator with the page number. Select '*No*' to deactivate the separator and have the plain catalog list.

Page Numbers Style - Choose the style of page numbers from the list.

These are the available page number styles:

• Button;

Short

Home > New Luma Yoga Collection

New Luma Yoga Collection

Shopping Options			
ACTIVITY	~	Page #1	
CATEGORY	\sim		
CLIMATE	~) Coming
COLOR	\sim		(L)
ECO COLLECTION	\checkmark		
ERIN RECOMMENDS	\checkmark		
FEATURES	\checkmark		
GENDER	~	Echo Fit Compression	Gwen Drawstr

Short

 \sim

• Text with divider;

MATERIAL

Home > New Luma Yoga Collection

New Luma Yoga Collection

Shopping Options		
ACTIVITY	\sim	Page #1
CATEGORY	\sim	
CLIMATE	\sim	
COLOR	\sim	
ECO COLLECTION	\sim	
ERIN RECOMMENDS	\sim	
FEATURES	\sim	
GENDER	\sim	Echo Fit Compression Gwen Drawst
MATERIAL	\sim	Short Short

• Text with background.

Home > New Luma Yoga Collection

New Luma Yoga Collection

Shopping Options			
ACTIVITY	\sim	Page #1	
CATEGORY	\sim		
CLIMATE	\sim		
COLOR	\sim		1 A
ECO COLLECTION	\sim		
ERIN RECOMMENDS	\sim		
FEATURES	\sim		
GENDER	\sim	Echo Fit Compression	Gwen Drawstr
MATERIAL	\sim	Short	Short

AJAX Progress Image Path - Insert the path to your own loading theme to customize your website. Choose what kind of loading wheels fits your web store layout.

Follow this detailed guide to customize the button and match it with the general webstore design.

AJAX Compatibility

You are able to add the compatibility with 3-rd party extensions which use AJAX. It can be done with the usage of the following sample:

<?php

<u>a</u>masty

```
class Test
{
    /**
     * @var \Magento\Framework\DataObjectFactory
     */
   private $dataObjectFactory;
    /**
     * @var \Magento\Framework\Event\ManagerInterface
    */
   private $eventManager;
   public function __construct(
        \Magento\Framework\DataObjectFactory $dataObjectFactory,
        \Magento\Framework\Event\ManagerInterface $eventManager
    ) {
        $this->dataObjectFactory = $dataObjectFactory;
        $this->eventManager = $eventManager;
   }
   public function execute()
    {
        $someHtml = '<div><img src="test.jpg">...</div>';
        $data = $this->dataObjectFactory->create(
            ſ
                'page' => $someHtml,
                11
\Amasty\PageSpeedOptimizer\Model\Output\LazyLoadProcessor::PAGE_CONFIG keys
                'pageType' => 'catalog category view',
                // optional. Example of configs
                11
\Amasty\PageSpeedOptimizer\Model\Output\LazyLoadProcessor::prepareLazyConfig
                'lazyConfig' => [
                ]
            1
        );
        //if Amasty PageSpeedOptimizer is not installed output page will be
the same
        $this->eventManager->dispatch('amoptimizer_process_ajax_page',
['data' => $data]);
        $someHtml = $data->getData('page');
        return $someHtml;
   }
}
```

When using AJAX scroll extensions, you can also convert images to WebP format without using Lazy Load. Please use the code below to achieve this:

Varnish Compatibility

To make Varnish compatible with the WebP functionality, you need to customize the Varnish configuration. Also, now the extension supports WebP for Safari 14.

To add WebP images support for users who have never visited the site using Varnish, you need to change the Varnish **defaul.vcl** configuration file.

Follow the steps below:

```
1. Find the vcl_hash subroutine
```

```
2. Add the code:
```

```
# Amasty Google Page Speed Optimizer device detection and cache separation
mechanism
    if (req.http.cookie !~ "X-Magento-Vary=") {
        call device_detect;
        if ((req.http.X-Amasty-Accept-Avif || req.http.X-Amasty-Accept-Webp)
&& req.http.X-Amasty-Device) {
        hash_data(req.http.X-Amasty-Accept-Avif);
        hash_data(req.http.X-Amasty-Accept-Webp);
        hash_data(req.http.X-Amasty-Accept-Webp);
        hash_data(req.http.X-Amasty-Device);
        }
    }
}
```

3. Add a new subroutine using this code:

Last update: 2025/07/30 magento_2:google_page_speed_optimizer https://amasty.com/docs/doku.php?id=magento_2:google_page_speed_optimizer 13:37

```
sub device detect {
    unset req.http.X-Amasty-Accept-Webp;
    unset req.http.X-Amasty-Accept-Avif;
    unset reg.http.X-Amasty-Device;
    if (req.http.Accept ~ "image\/webp" ||
        req.http.User-Agent ~
"(?i)(\biPhone|\biPod|\biPad|\bMacintosh).*Version\/(1[4-9]|[2-9][0-9])" ||
        req.http.User-Agent ~ "(?i)Edg|Firefox|Chrome|Opera"
    ) {
        set req.http.X-Amasty-Accept-Webp = "true";
    }
    if (req.http.Accept ~ "image\/avif" ||
        req.http.User-Agent ~ "(?i)(?!.*Edg)(Firefox|Chrome|Opera)" ||
        req.http.User-Agent ~
"(?i)(\bMacintosh).*Version\/(1[6-9].[4-9]|1[7-9]|[2-9][0-9])" ||
        req.http.User-Agent ~
"(?i)(\biPhone|\biPod|\biPad).*Version\/(1[6-9]|[2-9][0-9])"
    ) {
        set req.http.X-Amasty-Accept-Avif = "true";
    }
    if (req.http.User-Agent ~ "(?i)Mobile") {
        set req.http.X-Amasty-Device = "mobile";
    } elsif (req.http.User-Agent ~ "(?i)Tablet") {
        set req.http.X-Amasty-Device = "tablet";
    } else {
        set req.http.X-Amasty-Device = "desktop";
    }
}
```

4. Reload Varnish.

Additional packages (provided in composer suggestions)

To make additional functionality available, please install the suggested packages you may need.

Available in Pro and Premium tariff plans with no additional fees:

• amasty/page-speed-optimizer-with-speed-size - Install this module if you want to have the compatibility with the SpeedSize[™] image optimization service.

Available as a part of an active product **subscription** or support subscription:

For Pro and Premium versions:

- **amasty/module-page-speed-optimizer-hyva** Install this package to ensure the Page Speed Optimizer extension works with the Hyvä Theme.
- **amasty/module-avif-image-optimizer** Install this package to enable AVIF optimization support.
- **amasty/module-optimizer-pro-functionality** Install this package to enable Pro package features.
- **amasty/module-imagick-library-image-optimizer** Install this package for Imagick library compatibility.
- **amasty/module-optimizer-subscription-functionality** Install this package:
- 1. to enable the advanced preload functionality;
- 2. to choose a strategy for JavaScript optimization (defer/async);
- 3. to run the command for specific image(s) optimization;
- 4. to prioritize resource loading.

For **Premium** tariff plan only:

- **amasty/module-ajax-cart-hyva** Install this package to ensure Ajax Shopping Cart extension works with Hyvä Theme.
- amasty/module-optimizer-premium-subscription-functionality Install this package to access automatic optimization for existing images in folders and logging for the automatic image optimization process.

FAQ

- * Is your extension compatible with Varnish?
- * Does WebP work with Safari?
- * Can I create WebP copies for theme images?
- * How to optimize JS with the Google Page Speed Optimizer extension?
- * Why Webp images don't appear on the frontend?
- * When optimizing images, I encounter an error: libpng warning: Interlace handling should be turned on when using png_read_image. What can I do about it?
- * Show more articles \rightarrow

<u>a</u>masty

Last update: 2025/07/30 13:37

ADD TO CART

GET FREE CONSULTATION

×

Find out how to install the Google Page Speed Optimizer via Composer

From: https://amasty.com/docs/ - Amasty Extensions FAQ

Permanent link: https://amasty.com/docs/doku.php?id=magento_2:google_page_speed_optimizer

Last update: 2025/07/30 13:37

