

## Fastly IO Integration

Release		
Epic		
Status		
Team		

### 1. Objective:

1.1 Ibexa DXP customers could enable and configure Fastly IO on their portals to display images with improved page load time for various end-user devices.

### 2. Background:

Media websites display images everywhere on the website. These images are uploaded independently of various sizes for display based on-screen resolutions. These images consume high network bandwidth and increase the page rendering time on the end-user devices (browsers). Fastly Image Optimizer (IO) comes with the solution to bypass this manual work of image upload for multiple screen resolutions and offers a mechanism to reduce page load time to improve end-user experiences. The reduced page load time improves the overall website performance, boosts conversions, and contributes to Google search crawling.

In short Fastly IO: -

- i) Improve page load time: render the images on demand via their cloud edge platform by resizing, adjusting the quality, crop/trim, change orientations, convert formats and many more.
- ii) Reduce infrastructure costs: It reduces the third-party integration to render images. It stores the images at one origin and renders them based on the requests.
- iii) Delighted customer Experience: Optimize the page load time, improve performance, reduce operation costs.

### 3. Requirements

Ibexa Cloud enterprise plan comes with Fastly CDN package. By default, it's not enabled, enable it beforehand on the project as a prerequisite to configure Fastly IO. To configure Fastly IO, retrieve the Fastly service ID key (`$FASTLY_API_SERVICE`) and a fastly API token (`$FASTLY_API_TOKEN`) via platform CLI command for your hosting plan, and share it with the Fastly IO team to enable Fastly IO service on this plan by using the web UI on [fastly.com](https://fastly.com). To contact Fastly, create an account on [fastly.com](https://fastly.com) and contact their sales team for enabling fastly by sharing the Fastly Service ID.



#### Important note:

1. First configure Fastly IO on the staging environment, test it completely and then after enable on production.
2. It's recommended to contact Platform.sh to enable fastly IO over the chosen hosting plan by sharing the project ID.

Commented [BD1]: This is something Ibexa supports natively. This sentence makes is sound like this would be added my FIO.

- Platform.sh documentation says “All [Platform.sh](#) projects that have Fastly services attached to the production instance, also provide a separate test Fastly service as well, attached to your `staging` branch for testing.”

**Fastly IO on the website could be configured on the website via following:**

- Web UI of Fastly IO
- VCL upload on Fastly IO ([fastly.com](#))
- Ibexa DXP fastly IO bundle

**1. Web UI of Fastly IO:**

- Basic Configuration: Fastly IO website provides a simple intuitive interface to configure fastly IO. Follow step-by-step process as shown/explained in the fastly web interface UI.
- Image customization management: Fastly IO provides various options for optimizing/transforming the images as below

**Edit default settings**  
Image optimization settings guide

**Auto WebP?**  No (default) - Use the origin image file type or chosen format.  Yes - Deliver WebP images to supported browsers.

**Default WebP (lossy) quality**

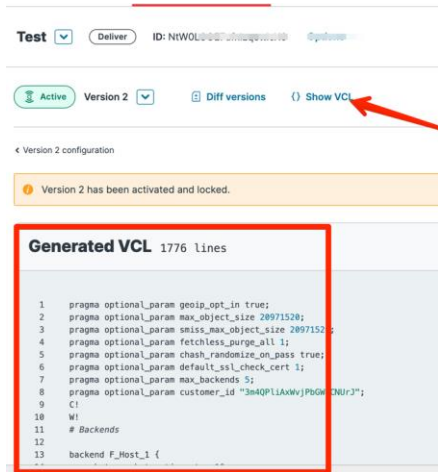
**Default JPEG format**  Auto (default) - Preserve the origin image JPEG format.  Baseline - Output the image when it's fully downloaded.  Progressive - Display the image using low-resolution first, then incrementally improve the quality as downloading continues.

**Default JPEG quality**

**Allow upscaling?**  No (default) - Recommended.  Yes - Allow upscaling.

**Advanced options** Enable Advanced DXP to Video, Resize View

Once the Fastly IO has been configured via Fastly IO web-interface UI, the entire configuration can be exported into the form of VCL file for an upload over Ibexa.Cloud Also, advance configuration can be managed directly into the VCL configuration file.



## 2. Direct VCL upload on Fastly IO (fastly.com)

Ibexa has developed a dedicated Fastly VCL (<https://github.com/ibexa/fastly>) which can be configured and optimized based on the requirements and can be uploaded directly via the web UI of fastly IO.

A successful integration of the fastly IO can be verified via two ways-

- i) Via PageSpeed Insight tests
- ii) and via Curl command.

### Via Curl command:

Run the command to check if fastly IO service is enabled or not  
 Command: `curl -sI "https://io.tacolabs.com/assets/tacos.jpeg?width=200"`

### Output:

```

content-type: image/jpeg
etag: "H3L4F2SmWG0jjiPbgoBWDiLaaabt9tmbJwZq2MQo/X0"
fastly-io-info: ifsz=98106 idlm=720x467 ifmt=jpeg ofsz=12494 odlm=200x130 ofmt=jpeg
fastly-stats: io=1
server: AmazonS3
x-amz-id-2: 49VVdw/m2icXkBrhqwIhSJWQ38mQd+0wRjHidc2b72kwb25uWl7K6L7ZKram5V74dDk62Bn/mU=
x-amz-request-id: GWRCPZVYT910VXCE
via: 1.1 varnish, 1.1 varnish
cache-control: no-store, max-age=0
  
```

## 3. Ibexa DXP fastly IO bundle

### Important note:

There is a module [“NovaeZFastlyImageOptimizerBundle”](#) developed by Noactive based on the Fastly IO API. Usage of this bundle for fastly IO implementation is out of scope.

## 4. User stories

US	Name	Title
----	------	-------

01	Enable & Configure Fastly IO	As a webmaster, I should enable & configure Fastly IO on my website to improve page-load performance.
02	Upload Fastly IO VCL	As a webmaster, I should be able upload configured fastly IO VCL directly on web UI of fastly.com so that I can improve the page-load time performance of my website.

## 5. Release plan

This integration has been planned in Version **4.4 (January 2023)** of the Ibexa DXP roadmap. In the coming days multiple meetings (Discovery and Kick-off) have been planned with the technical team members.

## 6. Success factors (via Fastly IO VCL upload)

- i) Webmaster should be able to enable/disable Fastly IO on their application.
- ii) Webmaster should be able to customize Fastly IO configuration on their application.
- iii) Webmaster should be able to check improved performance of page-load time of their application.
- iiii) Webmaster should be able to bypass Fastly IO implementation on certain media/image types of their application.
- v) Webmaster should be able to upload custom Fastly IO configuration (via VCL upload).

## 7. Open points:

### 7.1 Meeting (15/11/22)

Open points	Response (21/11/22)	Status
<ol style="list-style-type: none"> <li>1. Check fastly VCL config can be configured via git push.</li> <li>2. API to push/configure fastly IO Fastly VCL</li> <li>3. Get an environment with fastly CDN enabled</li> </ol>	<ol style="list-style-type: none"> <li>1. We will upload it directly to web UI.</li> <li>2. We will not use this solution.</li> <li>3. Waiting for P.sh action</li> </ol>	<ol style="list-style-type: none"> <li>1. Closed</li> <li>2. Closed</li> <li>3. In progress</li> </ol>

## 8. References

Name	Links
Fastly	<ul style="list-style-type: none"> <li>i) <a href="https://docs.fastly.com/en/fundamentals/fastly-cdn">https://docs.fastly.com/en/fundamentals/fastly-cdn</a></li> <li>ii) <a href="https://docs.fastly.com/en/fundamentals/fastly-io">https://docs.fastly.com/en/fundamentals/fastly-io</a></li> <li>iii) <a href="https://docs.fastly.com/products/image-optimizer">https://docs.fastly.com/products/image-optimizer</a></li> </ul>
Fastly on Magento	i) <a href="https://devdocs.magento.com/cloud/cdn/fastly-image-optimization.html">https://devdocs.magento.com/cloud/cdn/fastly-image-optimization.html</a>
Other links	<ul style="list-style-type: none"> <li>i) <a href="https://docs.platform.sh/domains/cdn/fastly.html">https://docs.platform.sh/domains/cdn/fastly.html</a></li> <li>ii) <a href="https://community.platform.sh/t/how-to-manage-your-own-fastly-configuration-custom-vcl/911">https://community.platform.sh/t/how-to-manage-your-own-fastly-configuration-custom-vcl/911</a></li> </ul>