Drupal + CDNs

Wim Leers
Senior Software Engineer at Acquia

Mail: wim.leers@acquia.com
Twitter: @wimleers
Terminology: page loading performance

- Resources: CSS, JS, images ...
- HTML

CDN usage: resources $\longleftrightarrow$ everything
CDN usage: resources $\leftrightarrow$ everything

Resources (step 1)

- Affects 90% of page load time
- Your app server still in control
- Easy, especially with Origin Pull
- Most benefit, minimal effort
## CDN usage: resources ↔ everything

### Resources (step 1)

- Affects 90% of page load time
- Your app server still in control
- Easy, especially with Origin Pull
- Most benefit, minimal effort

### Everything (optional step 2)

- Affects additional 10%
- Careful integration, less control
- Complex
- Maximal benefit, a lot of effort
## CDN usage: resources $\leftrightarrow$ everything

<table>
<thead>
<tr>
<th>Resources (step 1)</th>
<th>Everything (optional step 2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Affects 90% of page load time</td>
<td>• Affects additional 10%</td>
</tr>
<tr>
<td>• Your app server still in control</td>
<td>• Careful integration, less control</td>
</tr>
<tr>
<td>• Easy, especially with Origin Pull</td>
<td>• Complex</td>
</tr>
<tr>
<td>• Most benefit, minimal effort</td>
<td>• Maximal benefit, a lot of effort</td>
</tr>
</tbody>
</table>

In both scenarios: less requests to app server, i.e. positive impact
ESI: Drupal 8
ESI: Drupal 8

- “Blocks & Layouts Everywhere Initiative” (codename: SCOTCH)

ESI: Drupal 8

• “Blocks & Layouts Everywhere Initiative” (codename: SCOTCH)

• Drupal 8: Symfony’s HttpKernel component

  • Every part (block) of a page can be requested individually

  • Drupal 8 will use in-process subrequests to build a page

  • i.e.: Drupal itself will use ESI internally!

ESI: Drupal 8

• “Blocks & Layouts Everywhere Initiative” (codename: SCOTCH)

• Drupal 8: Symfony’s HttpKernel component

  • Every part (block) of a page can be requested individually

  • Drupal 8 will use in-process subrequests to build a page

  • i.e.: Drupal itself will use ESI internally!

• Result: ESI on a CDN becomes almost trivial (but not every CDN supports ESI)

Key Properties of a CDN

More: “Key properties of a CDN” — http://wimleers.com/article/key-properties-of-a-cdn
Key Properties of a CDN

• Geographical spread (PoPs)

More: “Key properties of a CDN” — http://wimleers.com/article/key-properties-of-a-cdn
Key Properties of a CDN

• Geographical spread (PoPs)

• Populating: Pull versus Push

<table>
<thead>
<tr>
<th></th>
<th>Pull</th>
</tr>
</thead>
<tbody>
<tr>
<td>transfer protocol</td>
<td>automatically</td>
</tr>
<tr>
<td>+</td>
<td>virtually no setup</td>
</tr>
<tr>
<td>—</td>
<td>no flexibility redundant traffic</td>
</tr>
</tbody>
</table>

More: “Key properties of a CDN” — http://wimleers.com/article/key-properties-of-a-cdn
# Key Properties of a CDN

- Geographical spread (PoPs)

- Populating: Pull versus Push

<table>
<thead>
<tr>
<th></th>
<th>Pull</th>
<th>Push</th>
</tr>
</thead>
<tbody>
<tr>
<td>transfer protocol</td>
<td>automatically</td>
<td>FTP, SFTP, rsync, WebDAV, Amazon S3 …</td>
</tr>
<tr>
<td>+</td>
<td>virtually no setup</td>
<td>flexibility no redundant traffic</td>
</tr>
<tr>
<td>-</td>
<td>no flexibility redundant traffic</td>
<td>setup</td>
</tr>
</tbody>
</table>

More: “Key properties of a CDN” — [http://wimleers.com/article/key-properties-of-a-cdn](http://wimleers.com/article/key-properties-of-a-cdn)
Key Properties of a CDN

- Geographical spread (PoPs)

- Populating: Pull versus Push

- Lock-in

<table>
<thead>
<tr>
<th></th>
<th>Pull</th>
<th>Push</th>
</tr>
</thead>
<tbody>
<tr>
<td>transfer</td>
<td>automatically</td>
<td>FTP, SFTP, rsync,</td>
</tr>
<tr>
<td>protocol</td>
<td></td>
<td>WebDAV, Amazon S3 …</td>
</tr>
<tr>
<td>+</td>
<td>virtually no setup</td>
<td>flexibility</td>
</tr>
<tr>
<td></td>
<td></td>
<td>no redundant traffic</td>
</tr>
<tr>
<td>-</td>
<td>no flexibility</td>
<td>setup</td>
</tr>
<tr>
<td></td>
<td>redundant traffic</td>
<td></td>
</tr>
</tbody>
</table>

More: “Key properties of a CDN” — http://wimleers.com/article/key-properties-of-a-cdn
Which CDN?
Which CDN?

- Performance
  - Low latency (geographical spread) — CDN’s POPs ↔ visitors
  - High throughput — important for large downloads, video streams

Which CDN?

• Performance
  
  • Low latency (geographical spread) — CDN’s POPs ↔ visitors
  
  • High throughput — important for large downloads, video streams

• Features
  
  • Type: Origin Pull vs. Push
  
  • e.g. automatic lossless image optimization, (realtime) statistics, authentication...

Which CDN?

• Performance

  • Low latency (geographical spread) — CDN’s POPs ↔ visitors

  • High throughput — important for large downloads, video streams

• Features

  • Type: Origin Pull vs. Push

    • e.g. automatic lossless image optimization, (realtime) statistics, authentication…

• Support & costs

Maximally exploiting a CDN for resources
Maximally exploiting a CDN for resources

- DNS prefetching
Maximally exploiting a CDN for resources

- DNS prefetching
- Auto-balancing over multiple CDN domain names
Maximally exploiting a CDN for resources

- DNS prefetching

- Auto-balancing over multiple CDN domain names

- “Far Future expiration”
  - Browser cache is faster than CDN
  - Mark files to expire years from now (“far future expiration headers”)
  - Unique file URLs
  - Cheaper!
Drupal CDN module

If you install the Advanced Help module, the CDN module will provide more and better help.

Status
- Disabled
- Testing mode
- Enabled

If you don't want to use the CDN to serve files to your visitors yet, but you do want to see if it's working well for your site, then enable testing mode. Users will only get the files from the CDN if they have the access files on CDN when in testing mode permission.

Display statistics
- Shows the CDN integration statistics of the current page, to users with the access per-page statistics permission.

Save configuration

Drupal CDN module

If you install the Advanced Help module, the CDN module will provide more and better help.

Mode

- Origin Pull
- File Conveyor

Choose a CDN integration mode.

MODE-SPECIFIC SETTINGS

CDN mapping

http://d87something714.cloudfront.net

Define which files are mapped to which CDN.
MODE-SPECIFIC SETTINGS

CDN mapping

http://d57something714.cloudfront.net

Define which files are mapped to which CDN.

☑ Far Future expiration

Mark all files served from the CDN to expire in the far future — improves client-side cacheability.
Note: this requires the "Aggregate and compress CSS files" performance setting to be enabled (or your site will break).

Unique file identifier generation

misc/*.modules/*:themes/*|drupal_version
sites/*|mtime
sites/*|.avi .m4v .mov .mp4 .wmv .flv|perpetual

Define how unique file identifiers (UFIs) are generated.

› AVAILABLE UFI METHODS

Save configuration

Drupal CDN module: when *not* to use it
Drupal CDN module: when not to use it

- When every millisecond matters (e.g. when you can’t cache pages)
- Designed for ease of use and front-end performance, not back-end performance
Drupal CDN module: when *not* to use it

- When every millisecond matters (e.g. when you can’t cache pages)
  - Designed for ease of use and front-end performance, not back-end performance
- When your site has a *very* complex CDN mapping
  - CDN module has `cdn_pick_server()` callback supports complex mappings
Measuring performance: SUM vs. RUM

Measuring performance: SUM vs. RUM

• Ideally: like Continuous Integration also Continuous Performance Monitoring!

Measuring performance: SUM vs. RUM

• Ideally: like *Continuous Integration* also *Continuous Performance Monitoring*!

• “Synthetic”: test script, only a few browsers, controlled environment

  • Not realistic

  • Great as reference point for tracking performance as site evolves

Measuring performance: SUM vs. RUM

• Ideally: like Continuous Integration also Continuous Performance Monitoring!

• “Synthetic”: test script, only a few browsers, controlled environment
  • Not realistic
  • Great as reference point for tracking performance as site evolves

• “Real”: actual visitors, all browsers, real-world environment
  • Very realistic
  • Ability to see in which specific location or browser site performance is good/bad

Measuring CDN performance impact — SUM

Measuring CDN performance impact — SUM

1. Configure a dev/staging app server

Measuring CDN performance impact — SUM

1. Configure a dev/staging app server

2. Ensure your pages’ resources are served from the CDN domain

Measuring CDN performance impact — SUM

1. Configure a dev/staging app server

2. Ensure your pages’ resources are served from the CDN domain

3. Perform a test with (the free & open source!) http://WebPageTest.org with a node that is far away from your origin (app) server

Measuring CDN performance impact — SUM

1. Configure a dev/staging app server

2. Ensure your pages’ resources are served from the CDN domain

3. Perform a test with (the free & open source!) `http://WebPageTest.org` with a node that is far away from your origin (app) server

4. Perform the same test, but use WebPageTest’s scripting engine to point to your origin server

Measuring CDN performance impact — SUM

1. Configure a dev/staging app server

2. Ensure your pages’ resources are served from the CDN domain

3. Perform a test with (the free & open source!) http://WebPageTest.org with a node that is far away from your origin (app) server

4. Perform the same test, but use WebPageTest’s scripting engine to point to your origin server

5. Compare results

Measuring CDN performance impact — RUM

Measuring CDN performance impact — RUM

1. Ensure your production site’s pages’ resources are served from the CDN domain, but only for e.g. 50% of your users

Measuring CDN performance impact — RUM

1. Ensure your production site’s pages’ resources are served from the CDN domain, but only for e.g. 50% of your users

2. Install “real user monitoring” (RUM) performance measurements: e.g. New Relic RUM or Torbit Insight

Measuring CDN performance impact — RUM

1. Ensure your production site’s pages’ resources are served from the CDN domain, but only for e.g. 50% of your users

2. Install “real user monitoring” (RUM) performance measurements: e.g. New Relic RUM or Torbit Insight

3. Compare results

Measuring CDN performance impact — RUM

1. Ensure your production site’s pages’ resources are served from the CDN domain, but only for e.g. 50% of your users

2. Install “real user monitoring” (RUM) performance measurements: e.g. New Relic RUM or Torbit Insight

3. Compare results

• Free measurements, but no analysis or UI — if you want to roll your own:
  • http://drupal.org/project/episodes
  • http://drupal.org/project/navigation_timing