CWP.

Broken File Links in CWP

Version 1.4

Table of Contents

Changes

Version	Author	Date	Comment
1.0	Ingo Schommer	06/03/2019	Initial release for DIA review
1.1	Ingo Schommer	07/03/2019	Clarified points about broken links in embedded content
1.2	Ingo Schommer	08/03/2019	Added section numbers Clarified broken links on replaced file contents (2.2) Clarified SEO impact (7.3) Clarified testing instructions (7.4)
1.3	Bryn Whyman	25/03/2019	Added specific CWP versions (4)
1.4	Bryn Whyman	09/04/2019	Added FAQ (7.6) Made reference to specific release (2.2.3)

CWP.

1. Document Purpose

This document is aimed at both technical and business roles within CWP agencies. It outlines a relatively complex issue around file linking behaviour in CWP which was uncovered in late 2018. The information should be used to make decisions on the impact of the identified issue in a particular agency context, and help to determine if and when to upgrade stacks.

2. For Stack Managers

2.1 Broken direct links to files after a CWP 2.x upgrade

If your CWP instance has upgraded from CWP 1.x to CWP 2.x, your instance likely has an issue with direct links to uploaded files. Direct links to uploaded files may no longer work because all files have changed their location in most upgrades. This affects direct links from external sources, as well as some links embedded in website content. There are no known issues around the data integrity of CWP-managed files.

2.2 Broken direct links to replaced files in CWP 2.x

If your CWP instance is on CWP 2.x, when replacing the contents of a file (e.g. through the CMS UI), direct links to that file are broken, as the file location changes with every replacement. This happens both on new sites starting on CWP2.x and sites which have been upgraded from CWP 1.x.

2.3 Next Steps

We will prepare a CWP 2.2.3 hotfix release in mid-March 2019, which will fix both issues by redirecting broken links. The next scheduled quarterly release (June 2019) will provide a more complete fix minimising those redirects, and fixing links embedded in website content. Depending on your impact assessment for your particular context, you can either decide to upgrade to this hotfix release, or defer the upgrade to the next scheduled quarterly release.

This is also a good opportunity to review if you have enabled the keep_archived_assets config setting, which retains deleted file content. This feature works in addition to the default behaviour of retaining metadata on deleted files, as well as retaining deleted file content via CWP's automated database and filesystem backup regime.

3. For Developers

Sites upgrading from CWP 1.x to CWP 2.x required a file migration task which moved file locations. While the new system added some features (draft and versioned files), it also caused a regression by default. Existing direct links to uploaded files no longer work because all files have changed their location. This limitation was pointed out in the upgrading advice to developers, but the impact wasn't sufficiently highlighted.

This also affects links to files and original size images embedded in website content prior to a CWP 2.x upgrade. It does not affect the majority of images which are resized when embedded into website content. It doesn't affect new sites built on CWP 2.x, or sites which opted into the *legacy_files=true* setting to retain the existing file locations.

CWP.

There are no known issues around data integrity. All files that were available on a CWP 1.x site are still available after upgrading to CWP 2.x. File contents, metadata, and versions migrated from CWP 1.x are retained at minimum in the daily backups for later retrieval. Older file versions created in CWP 1.x are still available in the database and filesystem, but aren't accessible through the CMS UI. Any versions created on the site after the migration to CWP 2.x are available through the CMS UI.

We will prepare a CWP 2.2.3 hotfix release in mid-March 2019, which will fix the broken links. A full fix is targeted for the next scheduled quarterly CWP recipe release in June 2019, which will progressively move files on creation or publication. An optional file migration task can be used instead to move them in one step.

4. Example

- Original file created under CWP 1.x: *assets/myfile.pdf*
- File migrated under CWP 2.x with legacy_filenames=false: *assets/[content-hash]/myfile.pdf* (Regression: links to *assets/myfile.pdf* no longer work)
- File migrated under CWP 2.x with legacy_filenames=true: *assets/myfile.pdf*
- File with updated file content under CWP 2.x: *assets/[new-content-hash]/myfile.pdf* (Regression: links to *assets/[content-hash]/myfile.pdf* no longer work)
- File with hotfix applied (CWP 2.2.3): *assets/[new-content-hash]/myfile.pdf* (redirects to *assets/myfile.pdf*)
- File with full fix applied (CWP 2.3.x) : *assets/myfile.pdf* (no redirect required)
- Newly uploaded file with full fix applied: *assets/my-other-file.pdf* (no redirect required)

5. Behaviour in CWP 1.x

- As part of the CWP 1.x recipe, the silverstripe/versionedfiles module provides version tracking of files and their content. It's enabled by default in the CWP recipe.
- There is no ability to upload files as a draft before publishing them (files are available for download directly after upload).
- The module deletes older file content when current file is deleted (but retains file metadata versioning).
- As part of the CWP 1.x recipe, the silverstripe/secureassets module provides restriction of files and folders to certain author groups (which can be used to emulate "draft files").
- Digital Records Act compliance is supported through this version tracking, as well as automated disk image level backups (which include database and assets).
- Disk image level backups are taken daily, and kept staggered for 7 years. Restores of individual files need to be requested as part of a disk image restore through the CWP Service Desk.

6. Behaviour in CWP 2.x

- SilverStripe Core features replace the silverstripe/versionedfiles and silverstripe/secureassets module functionality in the CWP 2.x recipe.
- SilverStripe retains version tracking of files and their contents.
- SilverStripe adds the ability to upload files as a draft before publishing (with authenticated access for draft files).
- SilverStripe provides a file migration task into the new data structure, which only covers currently published files, not their past versions from the versionedfiles module. These are available in the database, but not the CMS UI.

- The file migration task renames file paths and their links. While these links are automatically updated in HTML content on the next write (e.g. embedded images and linked documents), it breaks existing direct links to files. This regression is tracked in https://github.com/silverstripe/silverstripe-versioned/issues/177.
- At the point of upgrade, developers can choose to retain current file links (*legacy_filename=true*), which still provides version tracking of files and their content. But this option comes with a few documented tradeoffs: It does not provide tracking of replaced file contents, or draft states of replaced file contents. While there is a theoretical migration path to the new approach (*legacy_filename=false*), that path hasn't been scripted. Original file contents are still available through the disk level backups in CWP.
- Deletes older file content when the current file is deleted (but retains file metadata versioning). Provides an opt-in for keeping deleted files (*keep_archived_files=true*). Deleted file contents are still available through the disk level backups in CWP.
- No changes to disk image level backups processes or Digital Records Act compliance support.
- The developer upgrading instructions for CWP 2.0.0 link to the SilverStripe 4.0.0 instructions, which point out this behaviour, but not very clearly. Refer to the "New Asset Storage Mechanism" section: "Because the filesystem now uses the sha1 of file contents in order to version multiple versions under the same filename, the default storage paths in 4.0 will not be the same as in 3."

7. FAQ

7.1 Will this change affect my search engine ranking?

As long as your files are still linked on your website, search engines will pick up the new links on any projects which have already been migrated. The bugfix will redirect links, which passes on any SEO rankings to the new link location. Since links to files should be permanent after the bugfix has been applied, this can lead to improved search engine rankings (since existing files under new links don't need to be re-ranked by search engines).

7.2 Can I migrate away from the legacy_filenames=true option?

Technically yes, but there's no official migration script for it.

7.3 Can I still choose legacy_filenames=true when starting new upgrades?

Technically yes, but you need to be aware of the tradeoffs. Once the regression has been fixed, we don't see the need for people to choose this option any more.

7.4 How do I test that the fix has worked on my site?

After applying the hotfix, you can test that it applied correctly. Please perform these tests on a test or UAT environment, since your local environment might have different routing conditions.

1. Find an existing published and draft file. Get the link by clicking on the preview in admin/assets. It should link to *assets/[hash]/myfile.pdf*. For each of the files:
    a. Check that it still routes correctly (hashed links)
    b. Remove *[hash]/*, and check that it still routes correctly (hashless links)

You can also try this with new files created after applying the fix. In the next CWP quarterly release, we will provide a migration task moving to “hashless” file links by default (without redirects). Regardless of whether you choose to apply this migration task during the upgrade, both hashed and hashless links should continue to work.

7.5 How many people are affected by this?

Everyone upgrading a site to CWP 2.x or SS 4.x, unless they've actively evaluated the tradeoffs, and opted for *legacy_filenames=true*. Everyone with assets likely has this issue (since all of them could be externally linked). The impact of the issue depends on how strongly your project relies on those external links (e.g. externally maintained link lists from your agency’s intranet to your agency’s website, links embedded in published documents, etc.).

7.6 Will this patch redirect URLs for previous file versions?

This patch will redirect file URLs that have since been replaced by a more recent file version (and hash). For example, on installing this CWP 2.2.3 patch, links to a prior version of the same file: *assets/[old-content-hash]/myfile.pdf* will be redirected to the new file URL: *assets/[new-content-hash]/myfile.pdf.*