2025/09/16 19:31 1/3 Кэширование

Кэширование

«ДокуВики» ускоряет просмотр статей с помощью кэширования разобранных файлов¹⁾. Если существует актуальная скэшированная версия документа, то будет доставлена именно эта скэшированная копия вместо того, чтобы разбирать все данные заново. При редактировании и предпросмотре никакого кэширования не применяется.

Очистка кэша

Для принудительной очистки кэша отдельной статьи добавьте к её URL параметр purge. Например:

http://www.example.com/namespace/page?purge=true

Для принудительной очистки кэша всех статей, включая страницы инструкций, примените команду touch к локальному файлу конфигурации $conf/local.php^{2)}$.

Для принудительной очистки кэша только xhtml-страниц, примените touch k inc/parser/xhtml.php.

Двухстадийное кэширование

«ДокуВики» использует два файла кэша для каждой статьи. Для того, чтобы понять это, вы должны знать, что статья до того, как быть отображенной в XHTML, сначала разбирается в промежуточный формат инструкций. «ДокуВики» сохраняет в кэше оба представления — и инструкции и подготовленный XHTML.

• Кэш инструкций

Файл кэша инструкций подвергается обновлению только при изменений статьи, конфигурационных файлов³⁾, некоторых php-файлов «ДокуВики»⁴⁾ или при добавлении/удалении/включении/выключении плагина с помощью менеджера плагинов.

Кэш XHTML

На кэш XHTML кроме перечисленного выше влияет опция максимального возраста кэша (cachetime), собственные метаданные, некоторые другие php-файлы ⁵⁾, то существует или не существует цель внутренней ссылки, и то, что рассылка RSS требует обновления.

Размещение

Кэши XHTML и инструкций располагаются в директориях data/cache/ $*^6$. Файлы завершаются суффиксами .xhtml, .i. В директории кэша также хранятся и другие файлы:

- . feed различные рассылки RSS, сгенерированные для вики;
- .code порции статей между тэгами <code> ... </code> после применения синтаксической подсветки.

Плагины

Last update: 2022/07/21 10:32

Плагины теперь могут влиять на кэш через событие PARSER_CACHE_USE. Это позволяет плагинам, которые вносят дополнительные зависимости в определённые статьи, проверить эти зависимости и принудить «ДокуВики» обновить статью, если эти зависимости нарушены.

Замечание разработчику

До сих пор единственную возможность, которую имел разработчик для отключения кэширования, — это отключить его полностью. Это было необходимо для таких плагинов как «Discussion» и «Include».

Для того, чтобы использовать новую функциональность, синтаксические плагины должны:

- сохранять информацию, относящуюся к статьям, затрагиваемым ими, и зависимостям, специфичным для каждой из этих статей. Метаданные статьи являются удобным местом для этого. Доступ к метаданным может быть получен с помощью массива \$INFO['metadata'] и функций p_get_metadata() и p_set_metadata(). Если вы используете метаданные, пожалуйста, обратитесь к Dublin Core Metadata standards.
- добавить компонент плагина действий для работы с событием PARSER CACHE USE.

Сам кэш управляется с помощью кэширующего объекта⁷⁾. Основными частями этого объекта интересными для плагина являются:

- массив depends «ДокуВики» заполняет его всеми известными зависимостями статьи и затем использует стандартные процедуры для работы с ним. Плагины могут добавлять и изменять эти зависимости до того, как они обработаны. Различными видами зависимостей являются:
 - ∘ purge очистить кэш;
 - age очистить кэш, если он старее, чем назначенный возраст (зависящий от значения метаданных 'date valid');
 - files очистить кэш, если он старее любого файла, указанного в этом массиве. Могут быть добавлены только локальные файлы. Загляните в inc/cache.php за списком таких файлов. Возможно наиболее интересно, что скэшированный xhtml зависит от метаданных статьи.
- имя кэша, cache уникальный идентификатор, под которым храниться кэш. Обычно зависит от имени статьи, HTTP_H0ST и номера порта сервера. Плагины могут составлять более сложные идентификаторы, например, «Include» составляет идентификатор, используя имена включаемых статей и имеет или не имеет текущий пользователь права на чтение данных статей.

Индивидуальной очистка кэша отдельной статьи

Как описано выше, «ДокуВики» проверяет валидность кэша, а не активно сбрасывает кэш. Однако, xhtml-представление статьи зависит от метаданных. То есть, если файл метаданных более свежий, чем кэш, «ДокуВики» определит, что кэш невалиден, и перерендерит (пересоздаст) статью. Таким образом, мы можем обновить метаданные для того, чтобы сбросить кэш.

https://wwoss.ru/ Printed on 2025/09/16 19:31

```
/* code to expire the cached xhtml of page ns:page
 * $id = 'ns:page';
 * $data = array('cache' => 'expire'); // the metadata being added
 * $render = false;
                                         // no need to re-render metadata now
 * $persistent = false;
                                          // this change doesn't need to
persist passed the next metadata render.
 */
p_set_metadata($id, $data, $render, $persistent);
По поводу кэширования изображений см. статью «Работа с изображениями и медиафайлами».
He-UNIX-овые пользователи могут просто открыть этот файл и сохранить его заново — смысл
состоит в том, чтобы изменить метку времени изменения файла.
conf/dokuwiki.php и conf/local.php.
inc/parser/parser.php и inc/parser/handler.php/
inc/parser/xhtml & inc/parser/metadata
```

где * — односимвольное название директории, а data — настраивается в опции «savedir»

From:

https://wwoss.ru/ - worldwide open-source software

Permanent link:

inc/cache.php

https://wwoss.ru/doku.php?id=wiki:caching

Last update: 2022/07/21 10:32

