

COREMEDIA CONTENT CLOUD

Release Notes



Copyright CoreMedia GmbH © 2023

CoreMedia GmbH

Ludwig-Erhard-Straße 18

20459 Hamburg

International

All rights reserved. No part of this manual or the corresponding program may be reproduced or copied in any form (print, photocopy or other process) without the written permission of CoreMedia GmbH.

Germany

Alle Rechte vorbehalten. CoreMedia und weitere im Text erwähnte CoreMedia Produkte sowie die entsprechenden Logos sind Marken oder eingetragene Marken der CoreMedia GmbH in Deutschland. Alle anderen Namen von Produkten sind Marken der jeweiligen Firmen.

Das Handbuch bzw. Teile hiervon sowie die dazugehörigen Programme dürfen in keiner Weise (Druck, Fotokopie oder sonstige Verfahren) ohne schriftliche Genehmigung der CoreMedia GmbH reproduziert oder vervielfältigt werden. Unberührt hiervon bleiben die gesetzlich erlaubten Nutzungsarten nach dem UrhG.

Licenses and Trademarks

All trademarks acknowledged.

February 22, 2023 (Release 11.2204)

1. Preface	1
1.1. Audience	2
1.2. Typographic Conventions	3
2. Overview	4
3. Software License Agreement Changes	6
4. Release Information	8
4.1. Release 2204.3	9
4.1.1. Breaking Changes	10
4.1.2. Changes and Improvements	11
4.1.3. Fixed Issues	12
4.2. Release 2204.2	18
4.2.1. Breaking Changes	19
4.2.2. Changes and Improvements	21
4.2.3. Fixed Issues	24
4.3. Release 2204.1	28
4.3.1. Breaking Changes	31
4.3.2. Changes and Improvements	36
4.3.3. Fixed Issues	42
4.4. Release 2201.1	52
4.4.1. Breaking Changes	55
4.4.2. Changes and Improvements	60
4.4.3. Fixed Issues	65
4.5. Release 2110.1	75
4.5.1. Breaking Changes	87
4.5.2. Changes and Improvements	115
4.5.3. Fixed Issues	138
5. Upgrade Information	155
5.1. Upgrade to 2204.3	156
5.1.1. CoreMedia Core	156
5.1.2. CoreMedia Content Application Engine	156
5.1.3. CoreMedia Headless Server	157
5.2. Upgrade to 2204.2	158
5.2.1. CoreMedia Core	158
5.2.2. CoreMedia Studio Server	159
5.2.3. CoreMedia Workspace	159
5.2.4. CoreMedia Headless Server	160
5.2.5. Miscellaneous	161
5.3. Upgrade to 2204.1	163
5.3.1. CoreMedia Core	163
5.3.2. CoreMedia Studio Client	166
5.3.3. CoreMedia Studio Server	169
5.3.4. CoreMedia Workspace	170
5.3.5. CoreMedia Deployment	171
5.3.6. CoreMedia Connector for SAP Commerce Cloud	172
5.3.7. CoreMedia Headless Server	173
5.3.8. Miscellaneous	174
5.4. Upgrade to 2201.1	176
5.4.1. CoreMedia Core	176
5.4.2. CoreMedia Content Application Engine	178

- 5.4.3. CoreMedia Studio Client 180
 - 5.4.4. CoreMedia Workspace 182
 - 5.4.5. CoreMedia Connector for Salesforce Commerce Cloud 183
 - 5.4.6. CoreMedia Headless Server 183
 - 5.4.7. Miscellaneous 184
- 5.5. Upgrade to 2110.1 187
 - 5.5.1. CoreMedia Core 187
 - 5.5.2. CoreMedia Content Application Engine 203
 - 5.5.3. CoreMedia Studio Client 209
 - 5.5.4. CoreMedia Studio Server 214
 - 5.5.5. CoreMedia Workspace 218
 - 5.5.6. CoreMedia Frontend Workspace 227
 - 5.5.7. CoreMedia Deployment 228
 - 5.5.8. CoreMedia Commerce Hub 232
 - 5.5.9. CoreMedia Headless Server 239
 - 5.5.10. Miscellaneous 247
- 6. Support 259

1. Preface

This manual contains the release notes for all *CoreMedia Content Cloud* releases. You will find the latest version of this manual at the following URL: <https://documentation.coremedia.com/cmcc-11/release-notes>.

1.1 Audience

This manual is intended for everyone interested in the detailed changes between *CoreMedia Content Cloud* releases. It is specifically intended for developers and operators who want to update their CoreMedia system.

1.2 Typographic Conventions

This manual contains different fonts and types in order to label different elements. The following table lists typographic conventions for this documentation:

Element	Typographic format	Example
Source code	Courier new	cm systeminfo
Command line entries		
Parameter and values		
Menu names and entries	Bold, linked with	Open the menu entry Format Normal
Field names	Italic	Enter in the field <i>Heading</i>
CoreMedia Components		The <i>CoreMedia Component</i>
Entries	In quotation marks	Enter "On"
[Simultaneously] pressed keys	Bracketed in "<>", linked with "+"	Press the keys <Ctrl>+<A>
Emphasis	Italic	It is <i>not</i> saved
Buttons	Bold, with square brackets	Click on the [OK] button
Glossary entry	>>-shaped icon	>> LDAP
Code lines in code examples which continue in the next line	\	cm systeminfo \ -u user

Table 1.1. Typographic conventions

2. Overview

CoreMedia Content Cloud releases have one of the following types:

- **Advanced Maintenance Pack (AMP)** contain mostly bug fixes and minor improvements. AMPs are kept free of breaking changes, meaning that projects usually have only minor upgrade work to carry out, for example merging and regression testing.
- **Advanced Enhancement Pack (AEP)** also contain bug fixes and improvements. But they also include new features and enhancements, which might be breaking. Therefore, projects are often required to carry out upgrade tasks.

The very first AEP release of a new major version (for example, CoreMedia CMS 9) that is available to all customers, is also called the **General Availability (GA)** release.

The release overview site at <https://releases.coremedia.com/cmcc-11/overview/archive.html> provides an overview of all available release versions of CoreMedia Content Cloud.

Subscribe to the help center at <https://support.coremedia.com/hc/en-us/sections/360005297394-CoreMedia-Content-Cloud-Self-Managed> to receive notifications about release and product announcements.

The release notes of CoreMedia Content Cloud are sorted by the main modules of the CoreMedia system. Each section bundles details about a number of technical components as shown in the following table:

- **CoreMedia Core:** All changes related to the CoreMedia core components.
 - Content Server
 - Workflow Server
 - Content Application Engine [CAE]
 - Content Feeder
 - Importer
 - Site Manager
 - Search Engine
 - Command Line Tools
- **CoreMedia Studio:** All changes related to CoreMedia Studio
- **CoreMedia Elastic Social:** All changes related to Elastic Social
- **CoreMedia Adaptive Personalization:** All changes related to Adaptive Personalization
- **CoreMedia Frontend:** All changes related to the CoreMedia Frontend Workspace.

Release types

Available releases

Helpcenter

Module organization

Templates

Bricks

Themes

- **CoreMedia Headless Server:** All changes related to CoreMedia Headless Server
- **CoreMedia Workspace:** All changes related to the CoreMedia Workspace.
- **CoreMedia Deployment:** All changes related to the CoreMedia Deployment.
- **Miscellaneous:** All changes which cannot be assigned to a specific module.

3. Software License Agreement Changes

This notice is to inform you of some important changes to our software license agreement, beginning with the general availability of CoreMedia Content Cloud version 10 (CMCC 10).

CMCC 10 introduces new software add-on components that require a separate license, specifically:

- CoreMedia Content Hub
- CoreMedia Experience Feedback Hub
- CoreMedia Commerce Hub

The following components require a license for CoreMedia Commerce Hub:

- CoreMedia Connector for Salesforce Commerce Cloud
- CoreMedia Connector for SAP Commerce Cloud
- CoreMedia Connector for HCL Commerce
- CoreMedia Marketing Automation Hub

The following components require a license for CoreMedia Commerce Hub:

- CoreMedia Connector for Salesforce Marketing Cloud

For more details, see the file marked Schedule 1 Product Specification CMCC.

CoreMedia plans to release more Hubs and Connectors in the future and reserves the right to amend the software license agreement with regards to these new components and connectors.

Existing Customers

Existing CoreMedia customers may be entitled to the Commerce Hub component and connector at no additional cost. Additional connectors may need to be licensed.

A license for the first three adapters for the components Content Hub and Experience Feedback Hub is included in your existing software license agreement; additional connectors will require a license upgrade.

For full details regarding your contract please contact your CoreMedia account manager.

4. Release Information

This chapter contains all release notes for CoreMedia Content Cloud ordered by release version. Each release version shows the changes to its predecessor. Each subsection provides the following information:

- Notable highlights of the release.
- Breaking changes, which require action on your side. The necessary steps are described later in the upgrade section of this document.
- Changes and improvements that introduce new features or change behavior. These usually do not require any action.
- The issues that are fixed with this release.

4.1 Release 2204.3

The release contains:

- Breaking changes: 3
- Changes and improvements: 6
- Bugfixes: 65

Only issues which contain release notes are shown in this document. So, the numbers in the list above can be higher than the number of issues presented in the subsections.

Breaking Changes

- CMS-21210: [Cli Tool Argument Checks are Improved](#)
- CMS-21919: [Fix and Deprecate Method TransformImageService#getTransformationString](#)
- CMS-22091: [Third-Party Update: GraphQL-Java to 19.2](#)

Changes and Improvements

- CMS-21653: [Multi-Site Manual: Hints on CleanInTranslation](#)
- CMS-22014: [Changed CMS Version Compatibility Check For Plugins](#)

Bugfixes

- CMS-21995: [Third-Party Update: Postgresql](#)
- CMS-21980: [Third-Party Update: NGINX](#)
- CMS-21932: [Derive Site Robustness: Handle Links to Destroyed Contents](#)
- CMS-21773: [Documentation on How to Derive a Site](#)
- CMS-21516: [Irritating and unhelpful logging in EditorialCommentEventFetcher for timestamps with `getNano\[\]`](#)
- CMS-21408: [New ServerExport Option to Export Files in Lowercase](#)
- CMS-20981: [The Content Feeder Now Also Feeds String Lists And Considers Nested Structs](#)
- CMS-20106: [Cleaned up application-mls-local.properties](#)
- CMS-22058: [Removed pbe Taglib Declaration from Blueprint taglibs.jinc](#)

- CMS-22043: Fixed a Bug Regarding Annotation Lookup of Link Schemes
- CMS-22031: Fixed the Order of the 'viewEngines' Bean
- CMS-22011: Bugfix: Issue with Responsive Image Delivery for Extremely Small Images
- CMS-22001: Fixed the Handling of Preview Date
- CMS-21769: Fixed CSRF Documentation Bug
- CMS-19083: Fixed a Bug Regarding Encoding of Resource URLs
- CMS-22076: Fixed "Create From Template" Folder Initialization
- CMS-22068: Fixed Folder Suggestions for Content Rules
- CMS-21987: Workflow Window: Start Workflow Button is Disabled Until Validation is Finished
- CMS-21915: Fixed 'minSearchCharacters' of User Manager Search
- CMS-21885: Fixed Collapse State Handling of Form View and Preview
- CMS-21802: Fixed Memory Leaks in CMPictureForm and CMLImageMapForm.
- CMS-21781: Third-Party Update: CKEditor 4 to 4.19.0
- CMS-17961: Start Workflow Window: Start button is disabled during starting the workflow
- CMS-21671: Fixed Update of Start Translation Workflow Issues
- CMS-21505: Enhanced Studio-Server Support for Markup in Structs
- CMS-15645: Fixed Drag in LinkLists of Annotated LinkLists
- CMS-22082: Headless Server Library Update of Snake YAML
- CMS-21968: Expose id of CMPlaceholder
- CMS-21843: Headless won't Start with Deactivated P13N Extension
- CMS-21725: JavaDoc Updates: Null-Behavior for Struct List Properties in StructBuilder
- CMS-21881: Fixed Unified API Bug in Working Version References

4.1.1 Breaking Changes

CoreMedia Core

Cli Tool Argument Checks are Improved

Especially the `cm bulkpublish` command and other tools have not checked their parameter and arguments. In this specific case, a folder was passed to the `bulkpublish` command without the necessary `-f` option (because other tools do it that way). However, the tool did not complain about the unknown argument and simply assumed a default

value for the folder. This is now checked more closely for all tools that don't need further arguments and a usage message is printed. This change is only breaking if you have used the tools with wrong parameters. If you used them correctly, the behavior does not change.

[CMS-21210]

CoreMedia Content Application Engine

Fix and Deprecate Method `TransformImageService#getTransformationString`

The method `com.coremedia.cap.transform.TransformImageService#getTransformationString(...)` has been changed to return null for non-existing blobs instead of throwing a `NullPointerException`. Further the method has been marked as deprecated for removal because it was not intended to be public API.

[CMS-21919]

CoreMedia Headless Server

Third-Party Update: GraphQL-Java to 19.2

The library `com.graphql-java:graphql-java` was updated from 18.1 to 19.2. It contains several bugfixes and some security relevant improvements.

Follow [Section "Third-Party Update: GraphQL-Java to 19.2" \[157\]](#) for upgrade information.

[CMS-22091]

4.1.2 Changes and Improvements

CoreMedia Core

Multi-Site Manual: Hints on `CleanInTranslation`

`CleanInTranslation` is a scheduled task related to the CoreMedia Multi-Site feature that is executed in workflow server. It may happen, that this scheduled tasks struggles with the state of some localization workflows, which prevents clean-up of false-positive *in translation* states of some contents.

A manual section "*CleanInTranslation Challenges*" has been added to the Multi-Site Manual describing possible issues in detail, how to analyze and possibly fix them.

[CMS-21653]

Miscellaneous

Changed CMS Version Compatibility Check For Plugins

- The plugin property `plugin.requires` and the programmatic version check for Java Plugins in the CMS has been removed to fix a bug with the required cms version and to prevent further confusion.
- A new Plugin Schema has been released: <https://releases.coremedia.com/plugins/plugin-schema-3.0.0.json>.

The schema now requires a `minimum-cms-version`. The property is not verified automatically, but is used for documentation purposes and enables clients to implement such a check in their deployment processes.

[CMS-22014]

4.1.3 Fixed Issues

CoreMedia Core

Third-Party Update: Postgresql

Updated the postgresql driver to 42.4.1 to benefit from latest security bugfixes.

[CMS-21995]

Third-Party Update: NGINX

The NGINX docker image has been updated to the latest version 1.23.1-alpine to avoid security issues of its OS base image.

In an upcoming CMCC release we will switch to a rolling tag to get updates of the OS base image with every build. We recommended that you manually switch to a rolling tag like `1.23-alpine` to cope with the moving target of security vulnerabilities in the libraries of the underlying OS.

Among others, the update fixes: CVE-2022-32207 CVE-2022-28391 CVE-2022-27404 CVE-2022-28391 CVE-2022-1586 CVE-2022-1587

[CMS-21980]

Derive Site Robustness: Handle Links to Destroyed Contents

While links to destroyed contents should be prevented, actions such as *Derive Site* should be robust to handle them.

Derive Site has been hardened accordingly, which means:

- links to destroyed contents are kept as is (and should be fixed afterwards)
- links to destroyed master contents are rewritten as expected to the new master content

[CMS-21932]

Documentation on How to Derive a Site

Deriving a new localized site in Studio can lead to several issues when content of the master site is not managed carefully. A new section *Multi-Site Challenges | Deriving a Site* in the *Multi-Site Manual* gives recommendations on how to perform the task in order to achieve best results.

[CMS-21773]

Irritating and unhelpful logging in EditorialCommentEventFetcher for timestamps with 'getNano()'

Improved Timestamp Logging in `EditorialCommentEventFetcher` . Timestamps are now More Readable.

[CMS-21516]

New ServerExport Option to Export Files in Lowercase

Use option `--lowercase` to store the exported content using lowercase filenames. This option can be helpful to mitigate problems with non-case-sensitive file systems. The paths are kept unique by appending a hash to the stored file name. The converted path will not affect the reimport of the exported files.

[CMS-21408]

The Content Feeder Now Also Feeds String Lists And Considers Nested Structs

StructFeedablePopulator has been extended to also feed all keys, strings and string lists in nested structs or struct lists. Of course it is necessary that the respective content is re-indexed before these properties are searchable.

[CMS-20981]

Cleaned up application-mls-local.properties

The property `repository.url` and its helper property `content-management-server.host` in `application-mls-local.properties` have been superfluous ever since and have now been deleted.

[CMS-20106]

CoreMedia Content Application Engine

Removed pbe Taglib Declaration from Blueprint taglibs.jinc

Removed pbe taglib declaration from cae-base-lib's taglibs.jinc, since cap-editing-pbe-taglib-component has been removed as of 2110.1.

[CMS-22058]

Fixed a Bug Regarding Annotation Lookup of Link Schemes

Lookup of @Link Annotation failed under certain conditions. This bug is fixed.

[CMS-22043]

Fixed the Order of the 'viewEngines' Bean

The type of the 'viewEngines' bean was changed to 'LinkedHashMap', so that custom view engines can be correctly prepended or appended.

[CMS-22031]

Bugfix: Issue with Responsive Image Delivery for Extremely Small Images

Delivery of responsive images could fail, if the original image is extremely small and no crops had been defined in CoreMedia Studio. More specifically, this issue arose for images whose dimensions were smaller than the corresponding aspect ratio.

[CMS-22011]

Fixed the Handling of Preview Date

Fixed the handling of preview date parameter, so that the preview date parameter is ignored by the live CAE.

[CMS-22001]

Fixed CSRF Documentation Bug

The documented example of backward compatible CSRF tokens contained a bug that was fixed.

[CMS-21769]

Fixed a Bug Regarding Encoding of Resource URLs

Resource URLs containing non URL characters were encoded twice. This bug has been fixed.

[CMS-19083]

CoreMedia Studio Client

Fixed "Create From Template" Folder Initialization

Fixed issue where the base folders in the dialog have been initialized with wrong values when opened the first time.

[CMS-22076]

Fixed Folder Suggestions for Content Rules

Fixed an issue where folder names have not been accepted by the rule folder suggestion component.

[CMS-22068]

Workflow Window: Start Workflow Button is Disabled Until Validation is Finished

The start button of the workflow window for publication/translation was enabled when opening the window. For a large content set it might take some time until the validation finished. During the validation the button was usable but the usage of the button caused errors in the case the validation is not successful. Now the start button is disabled and will be enabled if the validation has been completed successfully.

[CMS-21987]

Fixed 'minSearchCharacters' of User Manager Search

If set to '0', the user manager now shows all users and groups immediately. This previously only worked with the workaround setting the value to '-1'.

[CMS-21915]

Fixed Collapse State Handling of Form View and Preview

The collapse state of form view and preview was not restored correctly, this is fixed now.

[CMS-21885]

Fixed Memory Leaks in CMPictureForm and CMLImageMapForm.

We fixed memory leaks in CMPictureForm and CMLImageMapForm.

[CMS-21802]

Third-Party Update: CKEditor 4 to 4.19.0

CKEditor 4 has been updated from 4.17.1 to 4.19.0. This also includes 4.18.0 which contains highly recommended security fixes.

[CMS-21781]

Start Workflow Window: Start button is disabled during starting the workflow

In the window for starting workflows [publication/translation] the start button was in some cases enabled after the start was triggered. If an author clicked the start button multiple times problems in the server occurred. Now the start button is always disabled while the workflow is starting.

[CMS-17961]

CoreMedia Studio Server

Fixed Update of Start Translation Workflow Issues

Fixed a bug in the start translation workflow window, which showed errors for dependent content items, even after all links to the problematic content items have been removed.

[CMS-21671]

Enhanced Studio-Server Support for Markup in Structs

Studio-Server may not have been able to deliver some Markup stored in Structs to Studio-Client, recorded in Studio-Server Log as `org.w3c.dom.DOMException: HIERARCHY_REQUEST_ERR`.

Studio-Server now can handle such Markup, which may be, for example, a result of importing pretty-printed `serverexport` data.

To work around this issue in previous versions, it is recommended not to use pretty-print option for `serverexport`, if these exported data are meant to be imported to CMS again.

[CMS-21505]

Fixed Drag in LinkLists of Annotated LinkLists

Fixed an issue where drops in linklists that have been used in annotated linklists did not work.

[CMS-15645]

CoreMedia Headless Server

Headless Server Library Update of Snake YAML

Snake YAML library was updated from `org.yaml:snakeyaml:1.29` to `org.yaml:snakeyaml:1.31`

[CMS-22082]

Expose id of CMPlaceholder

Expose the `CMPlaceholder` property `id` as `placeholderId` , since before it has been shadowed via the content `id`.

[CMS-21968]

Headless won't Start with Deactivated P13N Extension

Due to a refactored component scan, two xml resources were not discovered anymore when the P13N extension is deactivated, which in turn caused a failed headless server start. This bugfix reinserts the missing resources.

[CMS-21843]

CoreMedia Documentation

JavaDoc Updates: Null-Behavior for Struct List Properties in StructBuilder

The `StructBuilder` does not accept `null` entries for list properties. While the behavior did not change, the documentation got updated accordingly to emphasize this behavior.

[CMS-21725]

Miscellaneous

Fixed Unified API Bug in Working Version References

Fixed a bug in the Unified API when retrieving references to other content items for the working version of some content. Before the fix, methods `Version#getReferences()` , `Version#getReferencesMap()` , and `Version#getReferences(CapPropertyDescriptor)` could return outdated data for the special so-called "working version" [compare `Version#isWorkingVersion`].

[CMS-21881]

4.2 Release 2204.2

The release contains:

- Breaking changes: 5
- Changes and improvements: 13
- Bugfixes: 63

Only issues which contain release notes are shown in this document. So, the numbers in the list above can be higher than the number of issues presented in the subsections.

Breaking Changes

- CMS-21688: [Log Level Change for Publication Failure](#)
- CMS-21476: [Fixed Incorrect Defer Flag During Theme Import](#)
- CMS-21777: [Third-Party Update: Spring Boot and Related Dependencies](#)
- CMS-21765: [Picture Local Setting 'disableCropping' now Supported by Link Building](#)
- CMS-21612: [Fixed Prometheus Cache Metrics](#)

Changes and Improvements

- CMS-21719: [Third-Party Update: Apache Solr](#)
- CMS-21652: [Fixed Handling of Broken Internal Links in RichText](#)
- CMS-20733: [Deprecation: Bean Generator tool has been deprecated](#)
- CMS-18964: [Configured Less Strict Cookie Defaults for Local Development](#)
- CMS-21690: [Fixed 'minSearchCharacters' of User Manager Search](#)
- CMS-21721: [Third-Party Update: Apache Tika and Transitive Dependencies](#)
- CMS-21774: [Base Image Update](#)
- CMS-21766: [Headless Server now Sends Content-Disposition Header for Non-viewable Responses](#)
- CMS-21651: [Third-Party Update: graphql-java to 18.1](#)
- CMS-21390: [Third-Party Update : Maven Surefire](#)

Bugfixes

- CMS-21739: [Fixed ServerExport of Nested Blobs In Structs](#)

- CMS-18984: Improved CAE Local Properties for Template Development
- CMS-21884: Re-introduce RichTextPropertyFieldDelegatePlugin
- CMS-21717: Scripts and documentation now make sure that pnpm 6.x is used
- CMS-21641: Fixed Upload Settings "maxFileSize"
- CMS-21628: Fixed Library Performance Issue
- CMS-21626: Bad UX: Don't Show "Accepted" in Localization Nagbar is full of quirks
- CMS-21586: hideLabel Does not Always Work in PropertyFieldGroups.
- CMS-21471: RotateStyleClassPlugin Is Now Robust Against Being Destroyed While Active
- CMS-20621: Feedback Hub Comment Button Misplaced For Two Level Field Group Hierarchy
- CMS-21790: Content Hub Adapter YouTube Update
- CMS-19627: Fixed a bug that in some cases SASS files could be included multiple times
- CMS-21800: Image Hash Calculation Leads to Client-Side Caching Problems
- CMS-21669: MediaController now sets Content-Disposition Header
- CMS-21187: Changed Return Type of Authors Field to CMPerson
- CMS-21877: Third-Party Update: Jetty
- CMS-21809: Third-Party Update: org.json:json
- CMS-21624: Third-Party Update: Netty

4.2.1 Breaking Changes

CoreMedia Core

Log Level Change for Publication Failure

Publication failures were logged with level *WARN* in *Content Management Server* . This wrongly indicated an operational issue that would require action by the operational team, while it actually is a content issue that has to be handled by editorial staff. The log level has thus been changed to *INFO* . Feedback on publication failures in *Studio* and in command line tools remains unchanged.

Follow [Section "Log Level Change for Publication Failure" \[158\]](#) for upgrade information.

(CMS-21688)

CoreMedia Studio Server

Fixed Incorrect Defer Flag During Theme Import

As described in <https://documentation.coremedia.com/cmcc-11/artifacts/2201/webhelp/frontend-en/content/ThemeConfigReference.html> you can set the "defer" flag in your theme configuration to defer the execution of a JavaScript file. Due to a bug all CMJavaScript documents generated by the theme importer that were not meant to be deferred were configured in a way that caused the following HTML output:

Follow [Section "Fixed Incorrect Defer Flag During Theme Import" \[159\]](#) for upgrade information.

[CMS-21476]

CoreMedia Workspace

Third-Party Update: Spring Boot and Related Dependencies

Spring Boot has been updated to version 2.6.8. Related dependencies have been updated to at least the versions used by Spring Boot. This includes updates to fix reported security issues in Spring Security and Tomcat [CVE-2022-22976, CVE-2022-22978, CVE-2022-29885]. The new version of Spring Security contains a bug fix for certain configurations of password hashing [CVE-2022-22976], which may require additional upgrade steps if you have changed the configuration for password hashing in the *Content Server* or *Elastic Social*.

Follow [Section "Third-Party Update: Spring Boot and Related Dependencies" \[159\]](#) for upgrade information.

[CMS-21777]

CoreMedia Headless Server

Picture Local Setting 'disableCropping' now Supported by Link Building

The flag `disableCropping` in the local settings of a picture was not respected by the link building for the GraphQL property `uriTemplate`. This misbehaviour was fixed. Also, the delivery will now validate this flag. This means that uri with disabled cropping won't work when request the image with a cropName and width. As a consequence, this bugfix is considered breaking, as the validation may break previously working delivery image URLs!

Follow [Section "Picture Local Setting 'disableCropping' now Supported by Link Building" \[160\]](#) for upgrade information.

[CMS-21765]

Miscellaneous

Fixed Prometheus Cache Metrics

Spring Boot actuator endpoint `/metrics` provides several metrics on CoreMedia components [see *Operations Basics Manual* , section *Basics of Operation | Actuator Endpoints | Metrics Endpoint*]. In addition to that, actuator endpoint `/prometheus` publishes this data for scraping by monitoring application *Prometheus* . However, cache metrics were incomplete for some CoreMedia components when using this endpoint. With this fix, the complete set of cache information can now be retrieved from it.

Follow [Section "Fixed Prometheus Cache Metrics" \[161\]](#) for upgrade information.

[CMS-21612]

4.2.2 Changes and Improvements

CoreMedia Core

Third-Party Update: Apache Solr

Apache Solr has been updated to version 8.11.2.

For detailed changes of Apache Solr have a look at <https://solr.apache.org/news.html> and https://solr.apache.org/docs/8_11_2/changes/Changes.html .

For CMCC 11, a new coremedia/solr-base Docker image was released with tag 8.11.2, which is used in the Blueprint in `apps/solr/blueprint/solr-image/pom.xml` .

[CMS-21719]

CoreMedia Content Application Engine

Fixed Handling of Broken Internal Links in RichText

In case that an internal Link in RichText cannot be built because the LinkFormatter throws an exception, the live CAE used to discard the whole RichText. Now it just discards the A-Tag. Preview CAE still displays the exception.

[CMS-21652]

Deprecation: Bean Generator tool has been deprecated

Bean Generator tool is outdated and has been deprecated.

[CMS-20733]

Configured Less Strict Cookie Defaults for Local Development

The `local` spring profile for live and preview CAE now configures the following cookie defaults:

```
cae.cookie.force-secure=false
cae.cookie.same-site=Lax
```

[CMS-18964]

CoreMedia Studio Client

Fixed 'minSearchCharacters' of User Manager Search

- Fixed a documentation issue where the field *minSearchCharacters* had the default value *0* instead of *3*.
- Fixed user manager search behaviour when *minSearchCharacters* is set to *0*: When this field has value *0* now, all users or groups are shown by default.

[CMS-21690]

CoreMedia Workspace

Third-Party Update: Apache Tika and Transitive Dependencies

Apache Tika has been updated to version 2.4.1 to avoid security vulnerabilities CVE-2022-30126, CVE-2022-25169, and CVE-2022-33879. As part of this change, the following transitive dependencies of Apache Tika have been updated to match versions used by Tika.

Updated dependencies:

- Apache PDFBox 2.0.26
- Apache Tika 2.4.1
- ASM 9.3
- JHighlight 1.1.0

If you use these libraries in project code, please check their respective release notes for changes and upgrade information. No changes were necessary in the CoreMedia Blueprint for these updates.

[CMS-21721]

CoreMedia Deployment

Base Image Update

The base image of all Spring Boot applications has been updated to `coremedia/java-application-base:2.4.4-corretto-11`.

The update includes:

- update to Oracle Corretto `11.0.15`
- jmx exporter to `0.17.0`

see <https://hub.docker.com/r/coremedia/java-application-base> for details

[CMS-21774]

CoreMedia Headless Server

Headless Server now Sends Content-Disposition Header for Non-viewable Responses

For security reasons, HeadlessServer now sends a Content-Disposition HTTP header of type "attachment" with filename="api.json" for all requests to the GraphQL endpoints.

[CMS-21766]

Third-Party Update: graphql-java to 18.1

The library `com.graphql-java:graphql-java` has been updated to version `18.1`. The update contains several deprecations, that may affect custom code. Please see the original documentation of graphql-java for details.

[CMS-21651]

Miscellaneous

Third-Party Update : Maven Surefire

Maven Surefire has been updated to 3.0.0-M7.

[CMS-21390]

4.2.3 Fixed Issues

CoreMedia Core

Fixed ServerExport of Nested Blobs In Structs

Fixed a bug in the `cm serverexport` command-line tool, that no files were exported for nested blob properties that are defined inside struct properties.

[CMS-21739]

CoreMedia Content Application Engine

Improved CAE Local Properties for Template Development

The `local` spring profile for live and preview CAE now configures the following cookie defaults:

```
server.servlet.jsp.init-parameters.development=true
freemarker.template.update.delay.in.seconds=1
```

[CMS-18984]

CoreMedia Studio Client

Re-introduce RichTextFieldDelegatePlugin

The `RichTextFieldDelegatePlugin` was removed in CMCC 11, but is now reintroduced. It can now be found in the `@coremedia/studio-client.main.ckeditor4-components` package. It can be imported as follows:

```
import RichTextFieldDelegatePlugin from
```

```
"@coremedia/studio-client.main.ckeditor4-components/plugins/RichTextFieldDelegatePlugin";
```

[CMS-21884]

Scripts and documentation now make sure that pnpm 6.x is used

Although suggested by the pnpm client simply upgrading to pnpm 7.x leads to errors in the frontend workspace as well for the studio-client. If you have already upgraded to pnpm 7, please downgrade the used version by executing:

```
npm install -g pnpm@6
```

[CMS-21717]

Fixed Upload Settings "maxFileSize"

The upload setting `maxFileSize` can overwrite the existing maximum upload size of 64MB now. The size is valid for all upload fields.

[CMS-21641]

Fixed Library Performance Issue

Fixed issue of too many `NewContentActions` which slowed down the library significantly.

[CMS-21628]

Bad UX: Don't Show "Accepted" in Localization Nagbar is full of quirks

The behaviour of the `In Localization Nagbar` was changed: The Nagbar does not disappear anymore if the user accepts the mentioned workflow.

[CMS-21626]

hideLabel Does not Always Work in PropertyFieldGroups.

An issue with the `hideSingleComponentLabel` configuration in `PropertyFieldGroups` has been resolved: The `hideLabel` config now works for all group items.

[CMS-21586]

RotateStyleClassPlugin Is Now Robust Against Being Destroyed While Active

Before the change the timer wasn't properly cleaned up which could have led to many console errors.

[CMS-21471]

Feedback Hub Comment Button Misplaced For Two Level Field Group Hierarchy

Components like `StringPropertyField` and `RichTextFieldPropertyField` now provide the `sideButtonVerticalAdjustment` and `sideButtonHo`

`horizontalAdjustment` config options to reposition their `EditorialComments` `Button`.

[CMS-20621]

CoreMedia Studio Server

Content Hub Adapter YouTube Update

The bundled plugin Content Hub Adapter Youtube has been updated to version 2.0.3. For details see the change log of the plugin at <https://github.com/CoreMedia/content-hub-adapter-youtube/blob/v2.0.3/CHANGELOG.md>. This includes updates of transitive dependencies used by the plugin, for example library google-oauth-client was updated to avoid security vulnerability CVE-2021-22573 that was reported for the previous version.

[CMS-21790]

CoreMedia Frontend Workspace

Fixed a bug that in some cases SASS files could be included multiple times

A bug in the theme tool chain caused that the SCSS files of a brick were included multiple times in some cases. As the files were loaded in a way that these duplicates might also have been overriding other styling rules and this will no longer happen after fixing the bug this is considered a potentially breaking change. For that reason the fix is for now only enabled if the feature flag "enableDuplicateSassImportsFix" is set in the theme.config.json like this:

```
{
  ...
  "enableDuplicateSassImportsFix": true
}
```

[CMS-19627]

CoreMedia Headless Server

Image Hash Calculation Leads to Client-Side Caching Problems

When the changing a crop (e.g. clipping), the hash of the image did not change and as a consequence the URL also did not change anymore. This led to the problem that browser caches did not invalidate their caches. This bugfix fixes the hash calculation.

[CMS-21800]

MediaController now sets Content-Disposition Header

When downloading binaries via HeadlessServers media endpoint, the HTTP header `Content-Disposition` is now set, when using the media endpoints with a file-name.

Additionally the manual was complemented by the section 'MediaType Content Negotiation', explaining options to configure individually document types to suppress a Spring Web MVC default behaviour on potentially unsafe content types.

[CMS-21669]

Changed Return Type of Authors Field to CMPerson

Changed the return type of the authors field in CMTeasable to CMPerson. This is not breaking, since the previous type was CMTeasable, which is the superclass of CMPerson.

[CMS-21187]

Miscellaneous

Third-Party Update: Jetty

Jetty has been updated to version 9.4.48.v20220622 to avoid reported security vulnerabilities of previous versions.

[CMS-21877]

Third-Party Update: org.json:json

The Java third-party library org.json:json has been updated to version 20220320 to avoid reported security vulnerabilities of previous versions.

[CMS-21809]

Third-Party Update: Netty

Netty has been updated to version 4.1.77.Final to avoid security vulnerabilities of previous versions.

[CMS-21624]

4.3 Release 2204.1

The release contains:

- Breaking changes: 20
- Changes and improvements: 77
- Bugfixes: 180

Only issues which contain release notes are shown in this document. So, the numbers in the list above can be higher than the number of issues presented in the subsections.

Breaking Changes

- CMS-21491: [Class "TransformationService" Cleaned Up](#)
- CMS-21481: [New Method to Transform Image by "Image Transformation String" Directly](#)
- CMS-18001: [Unified API: ContentRepository.createContentBuilder\(\) Available](#)
- CMS-21355: [Runlevel Health Indicator Signals Wrong State in Runlevel MAINTENANCE](#)
- CMS-21279: [Changed SitesService Setup in cmd Tools](#)
- CMS-22223: [Moved thumbnailService and Helper Classes to cap-base-models](#)
- CMS-22220: [Third-Party Update: rxjs from 6.x to 7.x](#)
- CMS-21454: [Added Support for Multiple Bookmarks](#)
- CMS-21089: [Studio Client Unified API: Improved TypeScript Types](#)
- CMS-21144: [Fixed babel Configuration in studio-client](#)
- CMS-19126: [Enhancement: Custom Validation for Built-In Workflows](#)
- CMS-20445: [Third-Party Update: Spring Boot and Related Dependencies](#)
- CMS-21441: [Refactorings of the Traefik Configuration for the Developer docker-compose Setup](#)
- CMS-21079: [Fixed Directory Ownership in Container Images](#)
- CMS-21432: [Deleted Demo Content Site Apparel US](#)
- CMS-21256: [New Search Adapter with ExtensionPoint](#)
- CMS-21076: [FilterQueryArg now Part of Plugin Support Library](#)
- CMS-21063: [Expose SolrQueryBuilders as Beans for Plugins](#)
- CMS-21317: [MediaController Validates URL Parameters](#)
- CMS-19390: [Blob Upload Robustness](#)

Changes and Improvements

- CMS-21364: The management-tools Container can now be Configured to Redirect the Tools Output
- CMS-21111: Third-Party Update: Apache Tika and Transitive Dependencies
- CMS-21031: Content Server Actuator Endpoint to Suspend Blob Collector
- CMS-20166: Content-Feeder Health can be Monitored
- CMS-20022: Workflow Metrics
- CMS-19255: Automated Workflow Conversion
- CMS-17476: New Hints for Contents "In Localization"
- CMS-21502: DataviewFactory Metrics are now Exposed via Micrometer
- CMS-16537: Added new 'customizations' Actuator Endpoint to serve Spring Bean Customizations
- CMS-16536: Added new 'linkhandlers' Actuator Endpoint to Serve Spring Bean Link Handlers
- CMS-21415: Third-Party Update: Moment.js
- CMS-21397: Introduced New Search Nagbar
- CMS-21067: Added Member Refresh Button
- CMS-20979: Improved Preview Generation Label For Uploaded Files
- CMS-20978: Support of Catalog Objects Augmentation Using Multiple Selection
- CMS-20636: Translation/Synchronization: Support for Folder-Links in Dependent Content
- CMS-19529: Added Warning For Deleted Comments
- CMS-21110: Third-Party Update: MySQL
- CMS-21466: Third-Party Update: JSLT lib to Version 0.1.11 [was 0.1.8]
- CMS-20710: New Search Suggestions and ExtensionPoint
- CMS-21352: Adds pnpm Create Command
- CMS-21526: Third-Party Update: AWS SDK
- CMS-21281: Maven Dependency Management For Plugins
- CMS-20696: Sort Order of Catalog now Persistent in Thumbnail View
- CMS-16589: Faceted Search for Headless
- CMS-16540: Removed Artifact 'com.coremedia.cms:coremedia-service'.

Bugfixes

- CMS-21627: Third-Party Update : Drew Noakes Metadata Extractor
- CMS-21579: Third-Party Update: google-oauth-client
- CMS-21404: Info Actuator Endpoint Runs into Jackson Exception if Build Enables git-commit-id-plugin
- CMS-21360: motd and tools-context Function Missing in management-tools Container
- CMS-21105: Fixed Bug that Required Exact User Names for Login
- CMS-21065: Fixed NullPointerException in FeedableImpl
- CMS-20799: The License Check Now Considers IPs of all Network Interfaces
- CMS-20760: Changed User Manager Search
- CMS-20596: Documentation on Passwords in Tools dumpusers/restoreusers
- CMS-20536: The Garbage Collector is Now Robust Against Long Link Chains in Trash
- CMS-20535: Limit Content Server Threads Checking Database Connections
- CMS-20175: Fixed Parsing of Workflow Timer Element
- CMS-21402: coremedia.preview.js: Added Note About False Positive SAST Report
- CMS-21393: Search Query In Download Portal Is Now Escaped
- CMS-21346: Fixed Links to Categories with Apostrophes in IDs
- CMS-20774: Fixed Cache Capacity Handling and Reporting of DataViewFactory
- CMS-19526: Segment Validator Warns of Unfavourable Segment Names
- CMS-21582: TypeScript Handbook Is Now Properly Provided
- CMS-21513: Jangaroo npm Updated to Version 1.1.3
- CMS-21452: Fixed JavaScript in Taxonomy Manager
- CMS-21448: Fixed Referrer List Panel
- CMS-21438: Augmentation Folder Issue Fixed
- CMS-21399: raphael.js: Added Note About False Positive SAST Report
- CMS-21351: Fixed Layout of Additional Workflow Fields
- CMS-21255: Bug Fixed where Certain Domains Caused Error in Studio Login Screen
- CMS-21237: JavaScript Source-Maps for Some ExtJS Classes are no Longer Broken
- CMS-21167: Changed Editor Preferences for "Site Chooser"
- CMS-20723: State of MergeStrategyChooser is persisted in Studio
- CMS-19997: Fixed Tooltips For Pagegrid Layout Preview
- CMS-18177: Fixed Firefox Lazy Loading Problem

- CMS-21359: HideService Ignores ActionColumn
- CMS-21282: Third-Party Update: Liquibase
- CMS-21238: Fixed Disposing of Uploaded Temp Files
- CMS-19960: Changed Upload Dialog Behaviour
- CMS-19446: Fixed Possible "Undefined" Error in Notifications Menu
- CMS-19192: Fixed Library Selection
- CMS-19142: Change "contentType" Property on BlobPropertyField
- CMS-17814: Fixed Persona Checkout Issue
- CMS-17566: Studio-Apps Can Now Be Installed As App
- CMS-21645: Third-Party Update: Spring Framework and ASM
- CMS-21340: Fixed Security Issues
- CMS-21175: Preserve Order in Maps Returned by Headless Server
- CMS-21071: Handling DataFetcherResult in DataFetchers
- CMS-21553: Third-Party Update: ESAPI
- CMS-21453: Fix Catalog Repository Toolbar icon
- CMS-21319: Third-Party Update: Jackson
- CMS-21153: Third-Party Updates: gRPC-Java 1.44.1 and Netty 4.1.74.Final
- CMS-21150: Icon Fonts Should now Always be Up-to-date after Upgrading the Studio Client
- CMS-21148: Third-Party Update: commons-dbcp2
- CMS-21140: Third-Party Update: PostgreSQL JDBC Driver
- CMS-20627: Third-Party Update: ImageIO 3.8.2
- CMS-19910: Fixed Loading of Image Variants

4.3.1 Breaking Changes

CoreMedia Core

Class "TransformationService" Cleaned Up

Class TransformationService has been refactored to delegate image transformations to a new method in TransformImageService. As a consequence, obsoleted references to BlobTransformer and NamedTransformBeanBlobTransformer have been removed.

TransformationService's constructor as well as the corresponding Spring bean configuration in TransformationServiceConfiguration have been adjusted accordingly.

[CMS-21491]

New Method to Transform Image by "Image Transformation String" Directly

A new method `transformWithDimensions(Content content, String property, String operations):Optional<Blob>` was added to interface `TransformImageService` and its implementing class. With this new method, an image transformation can be requested via an Image Transformation String directly (see <https://documentation.coremedia.com/cmcc-11/articles/220/webhelp/aa-developer-en/content/imageTransformationAPI.html>).

Follow [Section "New Method to Transform Image by "Image Transformation String" Directly" \[163\]](#) for upgrade information.

[CMS-21481]

Unified API: ContentRepository.createContentBuilder() Available

A new way of creating contents within the Unified API is available:

Follow [Section "Unified API: ContentRepository.createContentBuilder\(\) Available" \[164\]](#) for upgrade information.

[CMS-18001]

Runlevel Health Indicator Signals Wrong State in Runlevel MAINTENANCE

When the Content Server is switched to runlevel MAINTENANCE either during startup or manually, the `health/runlevel` endpoint signals the server is UP. This makes it impossible to use the indicator for a readiness probe in Kubernetes, because the pod will be added to the services endpoint too early and UAPI clients that listen on a services availability would fail during the creation of the repository connection.

Follow [Section "Runlevel Health Indicator Signals Wrong State in Runlevel MAINTENANCE" \[165\]](#) for upgrade information.

[CMS-21355]

Changed SitesService Setup in cmd Tools

The `sitesService` bean is injected from `commandline-tools-context.xml` now instead of being registered programmatically. Now obsolete methods `registerSitesService` and `hasSingletonBeanOfType` in `com.coremedia.cmdline.SpringHelper` have been removed.

[CMS-21279]

CoreMedia Studio Client

Moved thumbnailService and Helper Classes to cap-base-models

This note covers changes to private API. Although private API is not subject to release notes and it can be changed without further notice we detected that this API was leaked in the blueprint so the code might have been copied or customized that is why we declared this change as possibly breaking.

Follow [Section "Moved thumbnailService and Helper Classes to cap-base-models" \[166\]](#) for upgrade information.

[CMS-22223]

Third-Party Update: rxjs from 6.x to 7.x

We now use the latest major version of rxjs in the studio-client workspace. In case you made any use of it in one of your packages make sure to upgrade rxjs as well. The changes can be seen here: <https://rxjs.dev/deprecations/breaking-changes>

[CMS-22220]

Added Support for Multiple Bookmarks

Inside the library, multiple content items can be bookmarked now. Additional properties have been added to `_Editor_properties.ts_` for this:

[CMS-21454]

Studio Client Unified API: Improved TypeScript Types

The part of Studio Client API that corresponds to CMCC's Unified API in Java (content, workflow, users) has been overhauled to provide more precise TypeScript types. Studio Client TypeScript code has been migrated from `ActionScript`, where it was not possible to describe callback function signatures and certain complex types (unions, intersections, generic types / type parameters). With TypeScript, now it is. To provide an improved formal description of the API, all vague types like `AnyFunction` or `any` have been replaced by more specific types, mostly already given in the textual documentation. This affects the packages `@coremedia/studio-client.client-core` and `@coremedia/studio-client.cap-rest-client`. Formally, changing the API to be more specific is a breaking change. But the required callback signatures, array element types and union types have already been documented as text, so if your code adheres these constraints, it is not going to break. Otherwise, only the TypeScript types of your code might have to be adapted, or the new, more precise types may even find hidden bugs in client code that need to be fixed.

[CMS-21089]

Fixed babel Configuration in studio-client

We fixed our internal babel configuration for the studio-client so code is no longer transpiled using the so called "loose" mode. This will cause that the resulting JavaScript code is now more precise regarding ES6 standards than before.

Follow [Section "Fixed babel Configuration in studio-client" \[168\]](#) for upgrade information.

[CMS-21144]

CoreMedia Studio Server

Enhancement: Custom Validation for Built-In Workflows

Custom start validators for built-in workflows (translation, synchronization, simple publication, and two-step publication) can now be configured via properties.

Follow [Section "Enhancement: Custom Validation for Built-In Workflows" \[169\]](#) for upgrade information.

[CMS-19126]

CoreMedia Workspace

Third-Party Update: Spring Boot and Related Dependencies

In order to benefit from the improvements of the latest version of Spring Boot and the Spring framework and to prepare for making use of the new features various 3rd party libraries have been updated.

Follow [Section "Third-Party Update: Spring Boot and Related Dependencies" \[170\]](#) for upgrade information.

[CMS-20445]

CoreMedia Deployment

Refactorings of the Traefik Configuration for the Developer docker-compose Setup

The Traefik configuration for the docker-compose developer setup has been refactored.

Follow [Section "Refactorings of the Traefik Configuration for the Developer docker-compose Setup" \[171\]](#) for upgrade information.

[CMS-21441]

Fixed Directory Ownership in Container Images

The ownership of the `/coremedia` directory was root but should be coremedia.

[CMS-21079]

CoreMedia Connector for SAP Commerce Cloud

Deleted Demo Content Site Apparel US

Demo content for site Apparel US was deleted because SAP Commerce does not provide a US site by default. The Apparel UK and Apparel DE still remain in demo content as example sites for the SAP Commerce integration.

[CMS-21432]

CoreMedia Headless Server

New Search Adapter with ExtensionPoint

The former `SearchServiceAdapter` and corresponding classes were replaced by the new `SearchAdapter`, featuring a SPI architecture and offering `SearchServiceProvider` as an `ExtensionPoint`. The default implementation in `DefaultSearchServiceProvider` implements the same behaviour as the former `SearchServiceAdapter`.

Follow [Section "New Search Adapter with ExtensionPoint" \[173\]](#) for upgrade information.

[CMS-21256]

FilterQueryArg now Part of Plugin Support Library

In order to enable customizations of the search via plugin extension points, it was necessary to move input type mode `FilterQueryArg` to plugin support. As a consequence the package has changed. Otherwise this change in nonbreaking.

[CMS-21076]

Expose SolrQueryBuilders as Beans for Plugins

In order to enable customizations of the search via plugin extension points, it was necessary to expose the so called `SolrQueryBuilders` as beans for plugins. As a consequence, `SolrQueryBuilder` had to be refactored to prevent circular bean dependencies with other search related extension points.

Follow [Section "Expose SolrQueryBuilders as Beans for Plugins" \[174\]](#) for upgrade information.

[CMS-21063]

MediaController Validates URL Parameters

The `MediaController` now validates the URL parameters 'id', 'propertyName', 'hash', 'cropName', 'width' and 'filename'. The missing validation was added for security reasons to prevent DOS attacks.

Follow [Section “MediaController Validates URL Parameters” \[174\]](#) for upgrade information.

[CMS-21317]

Miscellaneous

Blob Upload Robustness

Sometimes, CMCC client threads got stuck due to broken BLOB upload HTTP connections.

Follow [Section “Blob Upload Robustness” \[174\]](#) for upgrade information.

[CMS-19390]

4.3.2 Changes and Improvements

CoreMedia Core

The management-tools Container can now be Configured to Redirect the Tools Output

The management-tools container can now be configured to redirect the tools output either to a logfile per tool or to stdout. This can be configured using the `OUTPUT_REDIRECT` environment variable set to:

```
console [default] log - output will be written to /core  
media/tools/var/log/<TOOL>.out null - output will be suppressed
```

[CMS-21364]

Third-Party Update: Apache Tika and Transitive Dependencies

Apache Tika has been updated to version 2.3.0. As part of this change, the following transitive dependencies of Apache Tika have been updated to match versions used by Tika. This includes an update of Apache POI to fix security issue CVE-2022-2633.

Updated dependencies:

- Apache Commons IO 2.11.0
- Apache Log4J API and SLF4J Adapter 2.17.2

- Apache PDFBox 2.0.25
- Apache POI 5.2.2
- Apache Tika 2.3.0
- Apache XMLBeans 5.0.3
- BouncyCastle 1.70
- Curves API 1.07
- Glassfish JAXB 2.3.6
- Rome 1.18.0
- SLF4J 1.7.36

If you use these libraries in project code, please check their respective release notes for changes and upgrade information. No changes were necessary in the CoreMedia Blueprint for these updates.

[CMS-21111]

Content Server Actuator Endpoint to Suspend Blob Collector

The *Content Server* has a new Spring Boot actuator endpoint "blobcollector" that can be used to suspend (and resume) deletion of unused blobs at runtime.

Suspending blob deletion is a required step when making backups of non-transactional blobs stores like a file blob store. Previously, this required a change to configuration property `sql.store.collector.suspend` and a server restart. With the new actuator endpoint, no restart is required anymore. See section "Basics of Operation | Actuator Endpoints | Content Server Blob Collector Endpoint" in the Operations Basics Manual for details.

[CMS-21031]

Content-Feeder Health can be Monitored

The content-feeder is now added as an endpoint for health-check and can be monitored. It returns the state of the feeder.

[CMS-20166]

Workflow Metrics

Metrics for the number of open workflows are now available at the Spring Boot Actuator Metrics Endpoint. For details, see the Operations Basics Manual, section "Basics of Operation | Actuator Endpoints | Metrics Endpoint | Workflow Server Metrics".

[CMS-20022]

Automated Workflow Conversion

As an alternative to the workflowconverter tool (s. see section "Workflowconverter" in Workflow Manual), the conversion can now be done automatically by the workflow server. You can activate the automatic conversion by setting the application property `workflow.server.enable-workflow-converter` to `true`.

[CMS-19255]

New Hints for Contents "In Localization"

The Studio UI has been extended by hints in document forms (nagbars) when contents are part of an active localization process.

Similar to "Locked by Workflow" for publication processes, editors will get a hint "In Localization".

Possible implications, i.e., what editors facing this hint should be aware of, is described in Multi-Site Manual "Multi-Site Challenges" in section "Concurrent Editing Feature".

[CMS-17476]

CoreMedia Content Application Engine

DataviewFactory Metrics are now Exposed via Micrometer

DataviewFactory Metrics are exposed via DataViewFactory MBean some of which are provided as maps. When using the Prometheus `jmx_exporter` the maps are not exported to Prometheus. Thus, only accumulated metrics are available instead of the metrics per dataview class.

The dataview factory metrics are now exposed via Micrometer allowing Prometheus to receive the metrics per dataview class.

[CMS-21502]

Added new 'customizations' Actuator Endpoint to serve Spring Bean Customizations

Spring bean customizations are printed to the log upon application startup. There's now a new 'customizations' actuator endpoint serving the customizations report.

[CMS-16537]

Added new 'linkhandlers' Actuator Endpoint to Serve Spring Bean Link Handlers

Spring bean link handlers were printed to the log upon CAE startup. The logging on facility `com.coremedia.application.routing` was removed. There's now an actuator endpoint called 'linkhandlers' serving the resolved link handlers and link post processors.

[CMS-16536]

CoreMedia Studio Client

Third-Party Update: Moment.js

Updated Moment.js version to 2.29.2 to fix reported security issue [GHSA-8hfj-j24r-96c4](#)

[CMS-21415]

Introduced New Search Nagbar

When no search results have been found in the Studio library, the search will be executed again on the root folder and a nagbar is shown that the search results for this fallback are shown. This nagbar replaces the former "Retry Search in All Content" link, shown in the result list panel.

[CMS-21397]

Added Member Refresh Button

The user manager contains additional toolbar button to refresh members. These refresh button invalidate the selection from the UAPI cache, allowing to detect changes from the connected user provider.

[CMS-21067]

Improved Preview Generation Label For Uploaded Files

When files are dropped on the bulk upload dialog, a message is shown that a preview is generated. The message text has changed to emphasize that preview thumbnails are only generated for images.

[CMS-20979]

Support of Catalog Objects Augmentation Using Multiple Selection

The Studio allows now to augment a group a catalog objects of the same type(*Category* or *Product*) via multiple selection. The augmentation functionality will be active in the context menu and toolbar.

- Single catalog element are augmented with the same behavior as before.
- Multiple catalog elements are augmented by running the operation as *CoreMedia Studio Job* instead and the created contents are not automatically opened as a tab.

[CMS-20978]

Translation/Synchronization: Support for Folder-Links in Dependent Content

Prior to this fix, documents linking to folders (sometimes used in Settings-documents, for example), could not translated/synchronized from within CoreMedia Studio. A false-positive warning on missing document properties was raised.

With this fix, folders are supported in translation and synchronization workflows **as dependent content only** . Thus, you may safely translate/synchronize contents linking to folders.

Note, that these folders will not be listed within the workflow's detail view, as folders are considered implicitly created during a translation/synchronization workflow within that view.

[CMS-20636]

CoreMedia Studio Server

Added Warning For Deleted Comments

When a notification is clicked that points to a comment that has been deleted, a warning message is shown now, that this comment is not available anymore.

[CMS-19529]

CoreMedia Deployment

Third-Party Update: MySQL

MySQL was updated to version 8.0.28 in `global/deployment/docker/mysql/Dockerfile` .

[CMS-21110]

CoreMedia Headless Server

Third-Party Update: JSLT lib to Version 0.1.11 [was 0.1.8]

This is a drop in replacement maintenance update, containing non breaking bugfixes and new features only. For details please see original release notes at <https://github.com/schibsted/jslt/releases>

[CMS-21466]

New Search Suggestions and ExtensionPoint

The Headless server now offers suggestions for search expressions. Suggestions are retrieved via the new `SuggestionsAdapter` , where the business logic is implemented and replaceable by an SPI via the plugin `ExtensionPoint Suggestion SearchServiceProvider` .

Please see the headless manual for details.

[CMS-20710]

CoreMedia Documentation

Adds pnpm Create Command

Adds command for call of CoreMedia pnpm Starter Kit to manual.

[CMS-21352]

Miscellaneous

Third-Party Update: AWS SDK

In order to benefit from fixes for known bugs and security issues, the Amazon AWS SDK has been updated to version 2.17.169. There are no known issues or incompatibilities concerning CMCC.

[CMS-21526]

Maven Dependency Management For Plugins

The versions of Java third-party libraries Micrometer, Jakarta Validation API, and Hibernate Validator are now managed in the `common-thirdparty-for-plugins-bom` Maven artifact. Application plugins do not need to manage versions of these libraries themselves anymore, if they use these libraries with scope "provided" and import the above BOM Maven artifact.

[CMS-21281]

Sort Order of Catalog now Persistent in Thumbnail View

When the commerce catalog in the library is sorted by any of the columns in the list view, switching to the thumbnail view preserves the applied order.

[CMS-20696]

Faceted Search for Headless

HeadlessServer now comes with a new GraphQL query for a faceted search. See the headless manual for details.

[CMS-16589]

Removed Artifact 'com.coremedia.cms:coremedia-service'.

Removed the internal artifact 'com.coremedia.cms:coremedia-service'.

Versions of servlet container and java dependencies are now listed by the 'info' actuator if one of the 'dev' or 'local' spring profiles is active.

[CMS-16540]

4.3.3 Fixed Issues

CoreMedia Core

Third-Party Update : Drew Noakes Metadata Extractor

Third-party library `com.drewnoakes:metadata-extractor` has been updated to version 2.18.0 to avoid security vulnerabilities of the previous version.

[CMS-21627]

Third-Party Update: google-oauth-client

The third-party library `com.google.oauth-client:google-oauth-client` has been updated to 1.33.3 to avoid security vulnerability CVE-2021-22573.

[CMS-21579]

Info Actuator Endpoint Runs into Jackson Exception if Build Enables `git-commit-id-plugin`

If a build included the Git Commit ID (via enabling the `git-commit-id-plugin`), the actuator endpoint `info` ran into an exception for *Studio Server* and *CAE*. This has been fixed. The Git Commit ID will now be returned again properly.

[CMS-21404]

`motd` and `tools-context` Function Missing in `management-tools` Container

Due to the switch from openJDK to Amazon Corretto as a base image, there is also a switch from Debian to a CentOS base image and therefore the locations of these convenience scripts changed. This change fixes the behaviour when running with the Amazon Corretto base image.

[CMS-21360]

Fixed Bug that Required Exact User Names for Login

Fixed the bug that users from a specially customized user provider could not log in with a different identifier than the user name. For releases before 2007.1, custom user providers could return users for other login identifiers, but more recent releases prevented this. The possibility has been restored now, so that these custom user providers continue to work.

[CMS-21105]

Fixed NullPointerException in FeedableImpl

When the type of a collection set as feedable element cannot be determined, the type now correctly defaults to String.

[CMS-21065]

The License Check Now Considers IPs of all Network Interfaces

For IP based licenses all network interfaces are considered now and the check passes if there is at least one IP that can be successfully validated against the IP range configured in the license.

[CMS-20799]

Changed User Manager Search

The user manager does not load all users or group on initialization anymore. Instead an empty state panel is shown and a load is only triggered when more than `n` characters are inputted into the search field. The amount of characters can be configured via the Spring property `studio.usermanager.minSearchCharacters` and defaults to '3'.

[CMS-20760]

Documentation on Passwords in Tools dumpusers/restoreusers

Documentation of tools dumpusers/restoreusers has been extended to give additional information on handling of passwords.

[CMS-20596]

The Garbage Collector is Now Robust Against Long Link Chains in Trash

The Garbage Collector is now robust against long link chains in trash.

[CMS-20536]

Limit Content Server Threads Checking Database Connections

Fixed the bug that the number of threads used by the *Content Server* for checking database connections wasn't limited. This could have led to instabilities in cases, where many calls to the database are blocked due to some problems with the database. By default, the number of such threads is now limited to twice the maximum number of database connections. The limit can be changed with configuration property `sql.pool.check-connection-max-threads` as described in the Deployment Manual in section "CoreMedia Properties Overview | Content Server Properties | Properties for the Connection to the Database".

[CMS-20535]

Fixed Parsing of Workflow Timer Element

Timer elements in workflow definitions with relative values but without explicitly setting the property `relative` are now recognized correctly as 'relative' even for values > 999.

[CMS-20175]

CoreMedia Content Application Engine

coremedia.preview.js: Added Note About False Positive SAST Report

coremedia.preview.js might be reported by SAST tools like checkmarx. This is a false positive, the code injection is a central part of the preview integration and can only be performed by postMessage calls of authorized origins (see <https://documentation.coremedia.com/cmcc-11/artifacts/2201/webhelp/studio-developer-en/content/SecurityCSP.html>).

[CMS-21402]

Search Query In Download Portal Is Now Escaped

Before the change the search query could contain HTML which was added to the search result without being escaped .

[CMS-21393]

Fixed Links to Categories with Apostrophes in IDs

When rendering a Teaser in a fragment that has pointed to a category in Salesforce with an apostrophe in the ID the Velocity interpreter in Salesforce was crashing during the link building. That was affecting the whole page. Such apostrophes in link parameters will be "escaped".

[CMS-21346]

Fixed Cache Capacity Handling and Reporting of DataViewFactory

The DataViewFactory did not respect the configured cache capacities and fell back to the capacity of "java.lang.Object" for cache capacities whose keys did not represent the names of classes extending `com.coremedia.objectserver.dataviews.AssumesIdentity` or `com.coremedia.objectserver.dataviews.DataView` . This is the case for `com.coremedia.blueprint.common.contentbeans.Page` , for example. The DataViewFactory no longer maintains its own mapping of cache capacities but uses the capacities which are configured at the Cache bean. The DataViewFactory records the used cache classes and reports the corresponding cache capacities via JMX using the `CacheSizes` attribute or the `getCacheSizes` operation.

The cache size definitions from dataviews XML files have lower precedence than those defined at the cache. So if your project contains a `cachesize` entry for class `X` with value `Y` and your application config contains an entry `cache.capacities.X=Z` then the cache capacity is `Z` .

[CMS-20774]

Segment Validator Warns of Unfavourable Segment Names

It is not recommended to use a segment name in pages ending with a number preceded by a minus sign. The default request mapping can get confused by interpreting the number as content ID. In some rare cases that can lead to CAE handler error and such a page is not accessible. The ChannelSegmentValidator warns about such unfavourable segment names.

[CMS-19526]

CoreMedia Studio Client

TypeScript Handbook Is Now Properly Provided

The link in the README of the studio-client pointed to a dead end. This has been fixed now.

[CMS-21582]

Jangaroo npm Updated to Version 1.1.3

This fixes an issue when declaring a bindable Ext Config in TypeScript with a `#`-private field name with the same name as the public property *and* adding custom logic to the `set` accessor, like in this example:

```
interface MyComponentConfig extends ...,
  Pick<MyComponent, ... | "myConfig"> {
}

class MyComponent ... {
  #myConfig: number = 42;
  get myConfig(): number {
    return this.#myConfig;
  }
  set myConfig(value: number) {
    this.#myConfig = value;
    // any custom code, otherwise the accessor will be suppressed:
    console.log("gotcha!");
  }
}
```

The private field used to be rewritten to a public field with the same name, so that the line

```
this.#myConfig = value;
```

resulted in an infinite recursion at run-time.

Now, the private field is rewritten to use an `_` prefix like Ext does for Configs, so in the example, `_myConfig` would be used. As a workaround, you can use `#_myConfig`

directly in the TypeScript source code, but after the fix, the `#` -private field name without underscore should be best practice.

Furthermore, Jangaroo npm 1.1.3 updates the Jest version from 26 to 28 and it is now possible to customize the `jest.config.js` as the jangaroo build will no longer regenerate it.

[CMS-21513]

Fixed JavaScript in Taxonomy Manager

Fixed a JavaScript error that was triggered through a right-click in the taxonomy manager.

[CMS-21452]

Fixed Referrer List Panel

The `bindTo` expression for the referrer list panel allows `null` values now too.

[CMS-21448]

Augmentation Folder Issue Fixed

The display name of a categories (and its parent categories) is used as storage path below the Augmentations folder. If such a category name contained a space at the end the augmentation functionality was broken (nothing happened, the error was only visible in the browser console). These illegal characters are now removed from the storage path.

[CMS-21438]

raphael.js: Added Note About False Positive SAST Report

- First of all: the code is not reached by any of our usages
- Even if the part were reached the assignment to `innerHTML` actually is escaped (some tools do seem to detect it properly)
- The remaining two usages are assignments to actual properties of a DOM element which may not need to be escaped at all

[CMS-21399]

Fixed Layout of Additional Workflow Fields

The fields added as an additional workflow fields are now using the available width automatically - no additional configuration required.

[CMS-21351]

Bug Fixed where Certain Domains Caused Error in Studio Login Screen

We fixed a bug where domain names with certain character in it (e.g. `'` or `/`) caused an error in the Studio loading screen.

[CMS-21255]

JavaScript Source-Maps for Some ExtJS Classes are no Longer Broken

A recent third-party dependency upgrade did break the source-maps for some of the ExtJS classes. This has been fixed now.

[CMS-21237]

Changed Editor Preferences for "Site Chooser"

The settings property "Site Chooser" has been renamed to "Site Selector". It isn't possible to hide the locale string inside the menu's dropdown button anymore. Also, the "Show Site Name" setting has been replaced with "Show Site Selector" which enables the user to hide the site selector completely. The default is *false* for this setting.

[CMS-21167]

State of MergeStrategyChooser is persisted in Studio

The MergeStrategyChooser has a state handling now.

[CMS-20723]

Fixed Tooltips For Pagegrid Layout Preview

The tooltips for the iconized page grid preview has been rendered with *null* values. The tooltips are now rendering the placement names in this preview properly.

[CMS-19997]

Fixed Firefox Lazy Loading Problem

Fixed an issue where the lazy loading of a hug amount of content items sometimes did not work for Firefox.

[CMS-18177]

CoreMedia Studio Server

HideService Ignores ActionColumn

The HideService ignores the ActionColumn as the component offers items only as Array instead of ItemCollection. Now the use of ActionColumn doesn't interfere with HideService.

[CMS-21359]

Third-Party Update: Liquibase

Liquibase has been updated to version 4.8.0 to avoid security vulnerability CVE-2022-0839.

[CMS-21282]

Fixed Disposing of Uploaded Temp Files

When a blob is uploaded to Studio, it is converted to a temporary file, backing a temporary Blob object. This object/files is properly disposed/deleted now.

[CMS-21238]

Changed Upload Dialog Behaviour

The upload dialog is not modal anymore. This way, it is possible now to drop a target folder to the dialog from the docked library too.

[CMS-19960]

Fixed Possible "Undefined" Error in Notifications Menu

Fixed an issue where a possible undefined error lead to warnings in the console log.

[CMS-19446]

Fixed Library Selection

Fixed issue where items opened in the library are sometimes not selected.

[CMS-19192]

Change "contentType" Property on BlobPropertyField

According to the documentation, the field *contentType* for the *BlobPropertyField* allows to filter uploaded files for it's mime type. This field was only used the localization of the button label and the help text.

A new field *mimetypes* has been introduced which configures the valid mime types for the blob field in CSV format. The list of values may only contain primary mime types, such as "image" or "application".

[CMS-19142]

Fixed Persona Checkout Issue

Fixed an issue where personas are automatically checked out once they are opened in a tab.

[CMS-17814]

Studio-Apps Can Now Be Installed As App

When using the "install this site as an app" feature of Chrome-Based browsers caused the studio to not properly load. This has been fixed now.

[CMS-17566]

CoreMedia Workspace

Third-Party Update: Spring Framework and ASM

The Spring Framework has been updated to version 5.3.20 to avoid security vulnerabilities CVE-2022-22970 and CVE-2022-22971. ASM has been updated to version 9.3 to match the version used by Spring.

[CMS-21645]

CoreMedia Headless Server

Fixed Security Issues

Fixed Security Issues in JsonPreviewController.

[CMS-21340]

Preserve Order in Maps Returned by Headless Server

Previously the order of maps from structs were not preserved in the Headless Server. This was due to a bug in the implementation of the `{@MapToNestedMapsConverter }` which is now fixed. This affects all the maps retrieved from structs and the results of the `MapOf*` scalar types.

[CMS-21175]

Handling DataFetcherResult in DataFetchers

When the argument data of a DataFetcher (`environment.getSource()`) is wrapped in a DataFetcherResult, some DataFetchers did not resolve the value properly. This affected the `FilteringDataFetcher`, `ConvertingDataFetcher` and `CapStructPropertyDataFetcher`. The filter predicates `InProductionFilterPredicate` and `ValidityDateFilterPredicate` were not applied then, which led to unexpected visibility issues. This is fixed now.

[CMS-21071]

Miscellaneous

Third-Party Update: ESAPI

The third-party library `org.owasp.esapi:esapi` has been updated to version 2.4.0.0 to avoid security vulnerabilities reported for previous versions.

[CMS-21553]

Fix Catalog Repository Toolbar icon

Create product teaser icon on the catalog repository toolbar is displayed correctly

[CMS-21453]

Third-Party Update: Jackson

Jackson has been updated to version 2.13.2 and Jackson Databind has been updated to 2.13.2.1 to avoid security vulnerability CVE-2020-36518.

[CMS-21319]

Third-Party Updates: gRPC-Java 1.44.1 and Netty 4.1.74.Final

In order to benefit from fixes for known bugs and security issues, the gRPC-Java and Netty libraries have been updated to their latest versions:

- `io.grpc:*:1.44.1`
- `io.netty:*:4.1.74.Final`

[CMS-21153]

Icon Fonts Should now Always be Up-to-date after Upgrading the Studio Client

We added a timestamp to the URL the icon font is loaded from so new versions will always be updated after upgrading the Studio Client.

[CMS-21150]

Third-Party Update: commons-dbcp2

Apache commons-dbcp2 has been updated to version 2.9.0 to avoid a security issue of the previous version.

[CMS-21148]

Third-Party Update: PostgreSQL JDBC Driver

The PostgreSQL JDBC driver has been updated to version 42.3.3 to avoid security issues of the previous version [CVE-2022-21724, GHSA-673j-qm5f-xpv8].

[CMS-21140]

Third-Party Update: ImageIO 3.8.2

In order to benefit from the latest security improvements the third-party library ImageIO was updated to version 3.8.2

[CMS-20627]

Fixed Loading of Image Variants

Fixed an issue where image variants taken from the file `mediatransform.xml` have been appended to the list of content variants. These entries are not simply concatenated anymore, but variants taken from the XML file will be overwritten by content image

variants if they have the same name. By default, the image editor will sort the variants by name. If the default sort order (by name/ascending) should be changed, this can be done by using the utility class `ImageVariantsSorter.ts`. It can be changed to descending by applying

```
ImageVariantsSorter.VARIANTS_DEFAULT_SORT_ORDER =  
ImageVariantsSorter.SORT_TYPE.SORT_BY_NAME_DESC;
```

or by initializing the variable `ImageVariantsSorter.VARIANTS_DEFAULT_SORT_ORDER_BY_NAME` with a string array, which contains the name of the variants. If variant entries are missing in this list, they will be appended to result, sorted alphabetically.

(CMS-19910)

4.4 Release 2201.1

The release contains:

- Breaking changes: 20
- Changes and improvements: 70
- Bugfixes: 197

Only issues which contain release notes are shown in this document. So, the numbers in the list above can be higher than the number of issues presented in the subsections.

Breaking Changes

- CMS-20366: [Reduced Overhead for Background Task CleanInTranslation](#)
- CMS-16439: [Fixing Possibly False-Negative In-Translation State](#)
- CMS-20410: [New method TransformImageService#getTransformationString](#)
- CMS-20501: [The preview CAE now runs with standalone mode disabled](#)
- CMS-20471: [Static Files of External Preview moved to /static folder](#)
- CMS-20431: [Freemarker function bp.truncateText always returns un-encoded text](#)
- CMS-20963: [ToastManager has been changed to toastService](#)
- CMS-20732: [New Rule REST API](#)
- CMS-20652: [Strict Signature of Content Initializers](#)
- CMS-20580: [Third-Party Update: CKEditor 4 to 4.17.1](#)
- CMS-18728: [Studio Client Resource Bundle Changes](#)
- CMS-20989: [Master Version Texts Improved](#)
- CMS-20775: [Use new Maven Plugin to Handle CoreMedia Plugins](#)
- CMS-20575: [SFM C Extensions Extracted from Blueprint](#)
- CMS-19581: [Link Building for Augmented Pages \(commerce-led\)](#)
- CMS-20620: [Refactored Scalar FilterQueryArg to input type CustomFilterQuery](#)
- CMS-19110: [Added CacheControl support to Headless Server](#)
- CMS-20448: [Removed Usage of PowerMock Java Test Library](#)
- CMS-20367: [New ExtensionPoint CustomFilterQuery](#)
- CMS-20962: [Third-Party Update: mysql-connector-java](#)

Changes and Improvements

- CMS-20425: Third-Party Update: Apache Solr
- CMS-20376: JMX metrics for state of folder index
- CMS-19587: Link Parameters are passed into Commerce-Led Link Generation
- CMS-21054: Plugins for studio-client can now be Downloaded Behind a Proxy
- CMS-20683: LibraryTree declares drop events
- CMS-20619: Deprecation: OpenCalais
- CMS-20229: Added Form Locale Switcher
- CMS-19547: Added "Properties" Dialog for Folders
- CMS-17536: Unified Uploads
- CMS-20546: Preloading of child categories in Studio is configurable
- CMS-20434: Third-Party Update: Checker Framework
- CMS-21075: Update to coremedia/java-application-base
- CMS-20991: The java-application-base image has been updated
- CMS-21016: Updated commerce adapters to latest versions
- CMS-20711: Add plugin extension point to add Spring Converters to the headless server
- CMS-20327: Automatic conversion of input types to Java objects for adapters
- CMS-19709: Support for viewtype specific queries
- CMS-20982: Documentation for preloadChildCategories
- CMS-20645: Third-Party Update: log4j-to-slf4j
- CMS-19391: Enhancement in logging for UAPI
- CMS-19053: Cache Metrics for Spring Boot Actuator Metrics Endpoint

Bugfixes

- CMS-20984: Directory permissions and ownership corrected for management-tools image
- CMS-20899: Enhanced Robustness in Translation Workflows (Pre-Processing Stage)
- CMS-20768: cm cleanversions: Respect Versions Being in Translation
- CMS-20680: Start Translation Workflow Validation: May Report "No content to translate."
- CMS-20615: Third-Party Update : Netty

- CMS-20574: Improve Robustness and Error Messages for Translation Workflow Rollback
- CMS-20302: Updated documentation of extensions and their dependencies
- CMS-20137: Third-Party Update: NGINX
- CMS-19628: Third-Party Update: JDOM2
- CMS-19354: Fixed Content Feeder for slow content queries
- CMS-17243: "Force" Option for workflowconverter
- CMS-20776: Multi Catalog Preview
- CMS-20663: Fixed support for IE expressions in legacy template Page_additional-Head.ftl
- CMS-20610: Updated blueprint to support projects with only corporate-cae as extension
- CMS-20447: Fixed resolution of local message resources in developer mode
- CMS-19648: Fixed logging of CustomizeAnnotationBeanPostProcessor
- CMS-19458: Solved a Bug Preventing CAE Background Tasks to Access Contents Filtered by Validity Dates
- CMS-19259: Prevent Errors when Formatting Links from Download File Names
- CMS-18963: Fixed Context Path of External Preview
- CMS-20730: BindReadOnlyPlugin and AbstractBindEditablePlugin are no longer deprecated
- CMS-20647: Fixed: Richtext Differencing in Studio May Provide Invalid Results
- CMS-20611: WorkflowTransition#defaultNextTask is deprecated
- CMS-20517: Image Editor: Broken crops after change fixed
- CMS-20446: StringListPropertyField is once again editable
- CMS-20382: UI Enhancement: Contents Draggable from Translation Workflows
- CMS-20377: Updated Documentation
- CMS-19788: LargeOperationHelper is more robust in non-blueprint-setups
- CMS-19448: Content Hub Import: Show error when import fails
- CMS-21053: Content-Hub: Images broken
- CMS-20777: Multi Catalog Search
- CMS-20744: Content Hub Import: Class Cast Exception if ContentModel Contains List
- CMS-20699: Fixed Taxonomy Root Node Resolving
- CMS-20650: Third-party : h2database Dependency Removed
- CMS-20450: Prevented Wrong Segment Format Validation Issues
- CMS-20380: ContentHub UriBlobBuilder and Proxies

- CMS-20318: Fixed User Manager Delete Action
- CMS-20042: Better Login Check
- CMS-20010: Filter destroyed taxonomies when creating a comment
- CMS-20618: Third-Party Update : Logback
- CMS-20394: Timeout increased for TenantInitializerTest#tenantsRegistered
- CMS-21011: Improved Exception Handling of Exceptions during Validation Phase
- CMS-20600: Bugfix: Extending a GraphQL Enum causes a ClassCastException
- CMS-21009: Fixed a bug preventing apps to shutdown properly
- CMS-20961: Third-Party Update: Tomcat
- CMS-20686: Third-Party Update: Spring Boot 2.5.8
- CMS-20685: Updated Protocol Buffers for Java to 3.19.3
- CMS-20646: Third-Party Update : Jackson
- CMS-20592: Shared HCL/WCS Commerce Proxy enhancements
- CMS-20430: Root Category Preview
- CMS-20198: Images: Parameters in MIME types fixed
- CMS-19471: Fixed a bug preventing replacement of CAE richtext filter beans
- CMS-18634: Calista and Aurora UK site removed

4.4.1 Breaking Changes

CoreMedia Core

Reduced Overhead for Background Task CleanInTranslation

You may experience the Workflow Server's Background Task `CleanInTranslation` to cause considerable load on your CPU. We have addressed this issue with several adaptations. Some of them might be applied manually to existing systems already (for details see the end of this section). In addition to the updates below, `CleanInTranslation` got updated internally, providing an additional performance gain.

Follow [Section "Reduced Overhead for Background Task CleanInTranslation" \[176\]](#) for upgrade information.

[CMS-20366]

Fixing Possibly False-Negative In-Translation State

The scheduled job `com.coremedia.translate.workflow.CleanInTranslation` is, roughly speaking, responsible for garbage collecting meanwhile invalid "in-translation" states of contents due to aborted or escalated translation workflows.

Follow [Section "Fixing Possibly False-Negative In-Translation State" \[177\]](#) for upgrade information.

[CMS-16439]

CoreMedia Content Application Engine

New method `TransformImageService#getTransformationString`

`com.coremedia.cap.transform.TransformImageService` has been extended by a new method: `getTransformationString(...)` The method was introduced to enable future use cases of this class. You can use it to compute the transformation string for a given content, property, variant name, breakpoint width and height, and extension. If you have implemented this interface in a custom class, please make sure to implement this method as well. It is not used by default in the Blueprint, so you can theoretically leave the implementation empty.

[CMS-20410]

The preview CAE now runs with standalone mode disabled

The property `delivery.standalone` is now disabled by default in the preview CAE. As a result the preview CAE uses the same URLs as the live CAE and the default proxy configuration no longer requires `/blueprint/servlet` to be accessible when running behind a web proxy.

[CMS-20501]

Static Files of External Preview moved to `/static` folder

In order to simplify proxy rules to split requests to static files and dynamic servlets, the static files of the external preview have been moved to a static folder and are now served at

[CMS-20471]

Freemarker function `bp.truncateText` always returns un-encoded text

The Freemarker function `bp.truncateText()` lead to double encoded text, when used with markup, because the markup is first encoded by the function and secondly by Freemarker itself. Now, the encoding is left to do by Freemarker and the function returns un-encoded text for markup input as well as for strings. An implemented workaround with `"?no_esc"` is no longer needed and should be removed.

[CMS-20431]

CoreMedia Studio Client

ToastManager has been changed to toastService

The access to the *ToastManager* has been changed. It has been replaced by the global service *toastService*.

[CMS-20963]

New Rule REST API

With the introduction of the dialog "Folder Properties" a new REST API has been introduced for managing group and content permission via rules. Content.ts and Group.ts support the method *getRules* now. It returns a remote bean which contains a list of rule remote beans. These ones are displayed either in the Folder Properties dialog or on the rule tab of a user managers group tab.

Follow [Section "New Rule REST API" \[180\]](#) for upgrade information.

[CMS-20732]

Strict Signature of Content Initializers

The Content Initializer API has been stated more formally precise by specifying the initializer function signature as a TypeScript type `(content: Content) => void`. This concerns the following methods in `IEditorContext` (`@coremedia/studio-client.main.editor-components/sdk/IEditorContext`) and `IContentTypeContext` (`@coremedia/studio-client.cap-base-models/content/IContentTypeContext`):

Follow [Section "Strict Signature of Content Initializers" \[180\]](#) for upgrade information.

[CMS-20652]

Third-Party Update: CKEditor 4 to 4.17.1

CKEditor 4 has been updated to 4.17.1. It is included in `@jangaroo/ckeditor4`, version 1.1.0.

Follow [Section "Third-Party Update: CKEditor 4 to 4.17.1" \[181\]](#) for upgrade information.

[CMS-20580]

Studio Client Resource Bundle Changes

In the process of pulling features out of the editor-components module in order to make them available for the new Workflow app, many changes to resource bundles had to be made (these changes are in addition to the changes that happened due to the new

content type localization mechanism]. As the list is too extensive for a release note, we refer to [this link to a Coremedia Help Center article](#) .

[CMS-18728]

Master Version Texts Improved

The description texts above the master version property field have been improved. Existing texts have been removed from the *Editor.properties* resource bundle. For this reason, the ticket is marked as breaking.

[CMS-20989]

CoreMedia Workspace

Use new Maven Plugin to Handle CoreMedia Plugins

CoreMedia plugins are now synchronized by means of the `coremedia-plugins-maven-plugin` which handles adding, removing and downloading the respective plugins. The Groovy scripts in the `<workspace>-app/pom.xml` files (e.g. `cae/spring-boot/cae-live-app/pom.xml`) have been replaced with the `coremedia-plugins-maven-plugin` which supports caching of downloaded artifacts and respects Maven's proxy settings.

[CMS-20775]

SFMC Extensions Extracted from Blueprint

The extensions `sfmc` , `sfmc-p13n` and `sfmc-es-p13n` have been removed from Blueprint. The extensions for your version can be requested from the CoreMedia Support Team.

[CMS-20575]

CoreMedia Connector for Salesforce Commerce Cloud

Link Building for Augmented Pages [commerce-led]

For Augmented Pages (CMExternalPage) the external page ID and the external URI path can be set. If configured both, both values will now be rendered into the Json/VTL include template of the fragment response. Make sure your link rendering template on the commerce side can handle both parameters.

[CMS-19581]

CoreMedia Headless Server

Refactored Scalar `FilterQueryArg` to input type `CustomFilterQuery`

The former scalar `FilterQueryArg`, which was used to define the custom filter queries in a search query, was refactored to a GraphQL input type of the same name. In contrast to complex scalars, input types offer full support of auto completion and documentation while formulating a GraphQL query.

Follow [Section "Refactored Scalar `FilterQueryArg` to input type `CustomFilterQuery`" \[183\]](#) for upgrade information.

[CMS-20620]

Added `CacheControl` support to Headless Server

Headless Server now supports sending `CacheControl` headers for GET requests. Defaults can be configured using the `caas.cache-control.*` properties. The value of the the max-age directive is the minimum of the configured max-age value for the given request URL and the values of the `validFrom/validTo` properties of the requested contents. If no cache control configuration exists and the content does not contain a value for `validFrom/validTo` then no `CacheControl` header is sent. A negative max-age value indicates that no `CacheControl` header should be sent even in the presence of configured `validFrom/validTo` dates.

Follow [Section "Added `CacheControl` support to Headless Server" \[184\]](#) for upgrade information.

[CMS-19110]

Miscellaneous

Removed Usage of PowerMock Java Test Library

Previous releases used the PowerMock library in a small number of Java unit tests to mock static methods and object construction. These usages have now been replaced with new functionality of the Mockito testing library. PowerMock isn't used anymore.

Follow [Section "Removed Usage of PowerMock Java Test Library" \[184\]](#) for upgrade information.

[CMS-20448]

New `ExtensionPoint CustomFilterQuery`

The new release offers the new `ExtensionPoint CustomFilterQuery` in order to provide custom filter queries by a plugin.

Follow [Section “New ExtensionPoint CustomFilterQuery” \[185\]](#) for upgrade information.

[CMS-20367]

Third-Party Update: mysql-connector-java

The mysql-connector-java Java library has been updated in `shared/common/modules/shared/database-drivers/pom.xml` to version 8.0.28 to avoid security vulnerability CVE-2022-21363 of the previous version. Note, that the updated version of mysql-connector-java also dropped support for deprecated TLSv1 and TLSv1.1 connection protocols and requires using more-secure TLSv1.2 or TLSv1.3 protocols, which have to be supported by the used MySQL server.

Follow [Section “Third-Party Update: mysql-connector-java” \[185\]](#) for upgrade information.

[CMS-20962]

4.4.2 Changes and Improvements

CoreMedia Core

Third-Party Update: Apache Solr

Apache Solr has been updated to version 8.11.1. This includes a fix to avoid the Log4J security vulnerability CVE-2021-44228 in Solr.

For detailed changes of Apache Solr have a look at <https://solr.apache.org/news.html> and https://solr.apache.org/docs/8_11_1/changes/Changes.html.

For CMCC 11, a new coremedia/solr-base Docker image was released with tag 8.11.1, which is used in the Blueprint in `apps/solr/blueprint/solr-image/pom.xml`.

[CMS-20425]

JMX metrics for state of folder index

Two attributes were added to the `Server` management bean of the *Content Server*: `FolderIndexState` and `FolderIndexStateNumeric`. These attributes report the state of the folder index, which can be enabled using the `cm dbindex` tool and which speeds up server queries that are limited to subfolders of a given folder. Possible states are:

- `None` (0): Index has not yet been created.
- `Invalid` (1): Index is inconsistent.
- `Online` (2): Index is online.

- `Offline` [3]: Index is offline.
- `Unknown` [4]: Index state is unknown at the moment.

While `FolderIndexState` reports the descriptive string, `FolderIndexState` `Numeric` reports the numeric value.

[CMS-20376]

CoreMedia Content Application Engine

Link Parameters are passed into Commerce-Led Link Generation

Link parameters (e.g. "view") are passed into commerce-led link generation for commerce links. Template:

```
<#assign linkParams={ "name" : "peter", "city": "Hamburg" } >
<#assign testLink=cm.getLink(self, "niceView", linkParams)/>
```

Sfcc Link in Fragment Response:

```
<!--VTL
$include.url('CM-Content', 'contentId', '14992-14992', 'name', 'peter', 'city', 'Hamburg', 'view', 'niceView')
VTL-->
```

[CMS-19587]

CoreMedia Studio Client

Plugins for studio-client can now be Downloaded Behind a Proxy

The "http_proxy" and "https_proxy" configuration of NPM (see <https://docs.npmjs.com/cli/v8/using-npm/config#https-proxy>) is now automatically taken into account when downloading plugins during studio-client build.

[CMS-21054]

LibraryTree declares drop events

Since `LibraryTree` always adds a `ContentViewDragDropPlugin`, it now declares the additional events fired by that plugins as its own events. This allows you to attach listeners for these events in a type-safe way. This affects the events `beforeDrop` and `drop`. Without the fix, to add listeners for these event, you must use a type assertion `[as any]` or ignore the resulting type error `[// @ts-expect-error]`.

[CMS-20683]

Deprecation: OpenCalais

The OpenCalais integration for automatic tagging of content has been deprecated and will be removed with the next major release. For now, it has been disabled in the *TaxonomyStudioPlugin.ts*.

[CMS-20619]

Added Form Locale Switcher

Document formulators now have a locale switcher located on the toolbar of the form. It allows to switch between the different localized versions of the document and shows the version synchronization state.

[CMS-20229]

Added "Properties" Dialog for Folders

The context menu of the library tree and list view contains an additional entry "Properties" now. This entry is only visible for a folder selection and allows users to see details about the selected folder. If the user has supervise rights or is an administrative user, an additional tab is shown to modify the access permissions of the folder and configure supervise rights.

[CMS-19547]

Unified Uploads

The global or site specific configuration file "Upload Settings" or "UploadSettings" is responsible for timeout configurations now. If not configured, the default timeout of five minutes is used. This value may be overwritten via updating the variable `_UploadSettings.DEFAULT_TIMEOUT_MS_`.

[CMS-17536]

CoreMedia Studio Server

Preloading of child categories in Studio is configurable

All child categories are fully preloaded in Studio by default to determine if they are virtual or not. Set the property `studio.commerce.preload-child-categories` to `"ALL_EXCEPT_TOP_LEVEL"` if the top level categories shouldn't be preloaded when loading the root category. It can be useful if there is a huge number of top level categories and you are sure about that none of them are virtual. In commerce systems without a physical root category it can't be any other way. Moreover, if you are sure that there is no virtual category at all in the tree you can use the value `"NONE"`. In case a child category is not preloaded its virtual state is always false.

[CMS-20546]

CoreMedia Workspace

Third-Party Update: Checker Framework

The Checker Framework has been updated to version 3.15.0.

[CMS-20434]

CoreMedia Deployment

Update to coremedia/java-application-base

The base image `coremedia/java-application-base` has been updated to `2.4.2`. This update includes the JVM update to `11.0.14`.

[CMS-21075]

The java-application-base image has been updated

The base image of all Spring-Boot based services build by the blueprint has been updated to `coremedia/java-application-base:2.4.1-corretto-11`.

This update includes Amazon Corretto 11.0.14.

[CMS-20991]

CoreMedia Commerce Hub

Updated commerce adapters to latest versions

Updated the commerce adapters to the latest versions containing bug and security fixes:

- `commerce-adapter-mock` and Commerce Hub Base API: v2.0.6
- `commerce-adapter-hybris` : v2.0.6
- `commerce-adapter-sfcc` : v2.0.9
- `commerce-adapter-commercetools` : v2.0.5
- `commerce-adapter-wcs` : v2.0.7

Please see the release notes of the commerce adapters for more details.

[CMS-21016]

CoreMedia Headless Server

Add plugin extension point to add Spring Converters to the headless server

Add new plugin extension point to add one or more Spring `Converter` , to convert among other things the GraphQL input types which is needed when adding custom input types in plugins.

[CMS-20711]

Automatic conversion of input types to Java objects for adapters

The `graphqlConversionService` is a standard Spring `org.springframework.core.convert.ConversionService` enhanced by all instances of `org.springframework.core.convert.converter.Converter` created as regular Spring Beans.

Using this `ConversionService` together with the Spring Expression Language, adapters in the `content-schema` now may use typed parameters instead of a composition of collection objects deserialized when using GraphQL input types. If a suitable `Converter` is available, the `ConversionService` will invoke the conversion transparently.

[CMS-20327]

Support for viewtype specific queries

The Headless-Server now supports viewtype specific queries for `CMCollection` and `PageGridPlacement`. See documentation for details.

[CMS-19709]

CoreMedia Documentation

Documentation for preloadChildCategories

Added documentation for new configuration option `studio.commerce.preloadChildCategories` that was introduced with CMS-20546.

[CMS-20982]

Miscellaneous

Third-Party Update: log4j-to-slf4j

The Log4J library `log4j-to-slf4j` has been updated to version 2.17.1.

Note that recent security vulnerabilities for Log4J like CVE-2021-44228 ("Log4Shell"), CVE-2021-45046, CVE-2021-45105, and CVE-2021-44832 do not affect the usage of the Log4J adapter library `log4j-to-slf4j` that is used in CMCC to forward logging via SLF4J to Logback.

[CMS-20645]

Enhancement in logging for UAPI

A display in the log is added whenever there is an attempt to connect workflow server stating "Trying to connect to workflow server." In theme importer , connection to workflow server is removed. If the customer wants to establish the connection to workflow server, they can just change the `_repository.workflow.connect_` to true in application properties of theme-importer.

[CMS-19391]

Cache Metrics for Spring Boot Actuator Metrics Endpoint

Metrics for `com.coremedia.cache.Cache` usage are now available at the Spring Boot Actuator Metrics Endpoint. They are also still available via JMX as before. For details, see the *Operations Basics Manual* , section "Basics of Operation | Actuator Endpoints | Metrics Endpoint".

[CMS-19053]

4.4.3 Fixed Issues

CoreMedia Core

Directory permissions and ownership corrected for management-tools image

With this release, the management-tools image directory ownership for some directories will be fixed and set correctly to `coremedia:coremedia` . Affected directories are:

- `/coremedia`
- `/coremedia/tools`

In addition the following directories have been made volumes and are precreated so the tools can actually write to it, even when they are anonymous volumes.

- `coremedia/export`
- `coremedia/import`

[CMS-20984]

Enhanced Robustness in Translation Workflows (Pre-Processing Stage)

When translation workflows create contents in derived target sites, they may stumble across issues while doing so. Previously, this may have left created targets checked out by the artificial user `translation-workflow-robot`.

The processing in this pre-processing stage of the translation (and synchronization) workflow has been enhanced, so that at best effort, contents should not be left checked out by the artificial user.

Corresponding exceptions are still forwarded and will escalate the translation workflow. They should not be ignored, but investigated, as the intermediate content state may now be corrupted.

[CMS-20899]

cm cleanversions: Respect Versions Being in Translation

Prior to this fix, `cm cleanversions` may have removed versions of a master content which are referenced from an active translation or synchronization workflow. This may cause those workflows to escalate.

Now, `cm cleanversions` respects these so-called *Merge Versions*, which are those versions of a master content, which are *just about to be translated into some derived content*.

An update is strongly recommended, if you regularly use `cm cleanversions` with rather strict limits regarding `-keep-days` for example.

[CMS-20768]

Start Translation Workflow Validation: May Report "No content to translate."

When trying to start a translation workflow in CoreMedia Studio, it may report *No content to translate*, although contents have been selected.

This may happen, when you added contents from different sites or when you added a content which is not within a site.

This behavior is fixed now, and you will instead see the expected violation issues, such as:

- *There is a content item that does not belong to a site.*
- *There are content items that belong to different sites.*

[CMS-20680]

Third-Party Update : Netty

Netty has been updated to version 4.1.72.Final to avoid security vulnerabilities of the previous version.

[CMS-20615]

Improve Robustness and Error Messages for Translation Workflow Rollback

Translation Workflows are now more robust against errors during Rollback and generate more helpful log messages.

[CMS-20574]

Updated documentation of extensions and their dependencies

The section "Extensions and Their Dependencies" of the Blueprint manual has been updated. Before, the extension `es-alex` was listed in a wrong context.

[CMS-20302]

Third-Party Update: NGINX

The NGINX docker image that is used with CMCC has been updated to 1.21.5-alpine to avoid security issues of the previous version.

[CMS-20137]

Third-Party Update: JDOM2

JDOM2 has been updated to version 2.0.6.1 to avoid a reported security vulnerability of the previous version [CVE-2021-33813].

[CMS-19628]

Fixed Content Feeder for slow content queries

Fixed a bug in the *Content Feeder* that caused delayed index updates and blocked JMX calls, when it executed a long-running content query for less important changes that should be processed in the background.

[CMS-19354]

"Force" Option for workflowconverter

As an alternative to `-c`, you can now use `-f` in order to enforce conversion of process definitions. This is useful, if an LDAP group, which is used as role in a process definition, accidentally vanishes, reappears and gets a new CAP ID in the contentserver. The `workflowconverter` checks only for class incompatibilities but does not detect invalid group ids, and thus would not convert the workflow without the force option.

[CMS-17243]

CoreMedia Content Application Engine

Multi Catalog Preview

Commerce categories and products from a non-default catalog are now correctly opened in the preview using the correct catalog. This currently only affects our HCL Commerce integration out of the box, which supports multiple catalogs.

[CMS-20776]

Fixed support for IE expressions in legacy template Page._additionalHead.ftl

Fixed support for IE expressions in legacy template "Page._additionalHead.ftl" of "cae-base-lib" causing that no IE specific CSS/JS was loaded at all under certain conditions [e.g. if the "page" brick was not used].

[CMS-20663]

Updated blueprint to support projects with only corporate-cae as extension

Added the module `ec-cae-lib` as runtime dependency to `corporate-cae` to solve CAE startup failure on projects that ran the blueprint with only the corporate cae extension enabled.

[CMS-20610]

Fixed resolution of local message resources in developer mode

The loading of message resources from the local workspace [when `delivery.local-resources=true`] has been fixed. Like CSS and JS resources, they are now loaded from the extra resources directories [see configuration of `spring.boot.tomcat.extraResources`].

[CMS-20447]

Fixed logging of CustomizeAnnotationBeanPostProcessor

`CustomizeAnnotationBeanPostProcessor` sometimes erroneously logged that a bean customizer wasn't applied although it had been applied successfully.

[CMS-19648]

Solved a Bug Preventing CAE Background Tasks to Access Contents Filtered by Validity Dates

Bean `BlueprintHandlersCaeBaseLibConfiguration#validIdUntilConsumer` no longer throws an exception if the current request is not available when consuming a validity change instant.

[CMS-19458]

Prevent Errors when Formatting Links from Download File Names

The `CapBlobHandler` now returns a `UriComponentsBuilder` object instead of a `String` when building download links. This prevents error-prone guessing if provided links are URL encoded or not in the link formatter. Such errors could occur when `useCMDDownloadFilename=true` and the file name of a download contains both, encoded and unencoded characters (e.g. a `+` and a space).

[CMS-19259]

Fixed Context Path of External Preview

Resources for the external preview resources are loaded from a the "static" context path. The optional "servlet" segment is removed from path.

[CMS-18963]

CoreMedia Studio Client

BindReadOnlyPlugin and AbstractBindEditablePlugin are no longer deprecated

Actually the deprecation was a mistake, the given replacement has never made it into a release.

[CMS-20730]

Fixed: Richtext Differencing in Studio May Provide Invalid Results

Due to an issue in data-processing of CKEditor 4, you may have experienced invalid change annotations in differencing view for CoreMedia Richtext properties. One observed effect: Sections were marked as deleted, which never were deleted.

A workaround got applied in data-processing, which ships with CoreMedia Studio.

If you rely on differencing results, an update is strongly recommended.

[CMS-20647]

WorkflowTransition#defaultNextTask is deprecated

`WorkflowTransition#defaultNextTask` is deprecated and will be removed in the future. Please use the new config `defaultNextStep` instead.

[CMS-20611]

Image Editor: Broken crops after change fixed

In some cases resizing crops were broken by a recent bug fix. Now all crops can be resized again.

[CMS-20517]

StringListPropertyField is once again editable

As the title says items added to a StringListPropertyField can once again be edited.

[CMS-20446]

UI Enhancement: Contents Draggable from Translation Workflows

Contents, displayed in translation workflows, can now be dragged and dropped to different areas of the Studio.

[CMS-20382]

Updated Documentation

Removed documentation about deleted public API method *EditorContext#getEnabledSearchFilterIds* .

[CMS-20377]

LargeOperationHelper is more robust in non-blueprint-setups

In non-Blueprint setups, copy/paste operations in the library could fail due to the LargeOperationHelper. This is fixed now.

[CMS-19788]

Content Hub Import: Show error when import fails

When an import terminates with an exception the import will now be visualized as failed instead of successful.

[CMS-19448]

CoreMedia Studio Server

Content-Hub: Images broken

Due to a bug in JVM version 11.0.14.9 the content-hub images (pictures) could not be loaded. The container base image `coremedia/java-application-base` has been updated to JVM version 11.0.14.10.

[CMS-21053]

Multi Catalog Search

Search within commerce products from a non-default catalog is now correctly executed. This currently only affects our HCL Commerce integration out of the box, which supports multiple catalogs. The fix requires HCL Commerce Adapter version 2.0.7 or higher.

[CMS-20777]

Content Hub Import: Class Cast Exception if ContentModel Contains List

To import a ContentHubItem the ContentModel can specify which properties has to be set in the imported Content by setting ContentModel.contentProperties. If the ContentModel.contentProperties contains a list of ContentModelReference the references are resolved and imported as Content too. Unfortunately a for Lists other than ContentModelReference a ClassCastException has been thrown and the import failed. This behavior has been resolved and Lists other than ContentModelReference stay untouched and will be copied as Content property to the imported Content.

[CMS-20744]

Fixed Taxonomy Root Node Resolving

The resolving of taxonomy root nodes is now implemented with a content query. This way, nodes are not skipped because of the SOLR search limit.

[CMS-20699]

Third-party : h2database Dependency Removed

h2database dependency raised issue in security scans and since this dependency was used only for testing purpose, it has been removed.

[CMS-20650]

Prevented Wrong Segment Format Validation Issues

The `SegmentFormatValidator` now makes use of `UrlPathFormattingHelper#tidyUrlPath` before the actual validation takes place. This ensures that the validation is based on the same segment formatting that is used in the deliveries and thus prevents false-positive validation issues.

[CMS-20450]

ContentHub UrlBlobBuilder and Proxies

The UrlBlobBuilder (Studio-Server API for Content Hub) now respects a possibly set authenticator via `java.net.Authenticator#setDefault`. This, for example, provides the opportunity, to enable password authentication, if Studio server runs behind a proxy requiring authentication.

[CMS-20380]

Fixed User Manager Delete Action

Admin users are no longer able to delete themselves when logged into the Studio.

[CMS-20318]

Better Login Check

The login now checks the permissions of the given user more detaily.

[CMS-20042]

Filter destroyed taxonomies when creating a comment

When creating a new comment, an exception occurred, if the linked taxonomy was deleted or destroyed. This is fixed now, the affected taxonomies are filtered out.

[CMS-20010]

CoreMedia Workspace

Third-Party Update : Logback

Logback has been updated to version 1.2.10 to avoid security vulnerabilities of the previous version.

[CMS-20618]

Timeout increased for TenantInitializerTest#tenantsRegistered

The timeout for TenantInitializerTest#tenantsRegistered has been increased. Before, the test could fail under high load.

[CMS-20394]

CoreMedia Headless Server

Improved Exception Handling of Exceptions during Validation Phase

Exceptions during the validation phase of a GraphQL query resulted in a HTTP 500 error for a validation error, e.g. "maximum query depth exceeded". This was caused by an incorrect exception handling in the `PreparedDocumentProvider` of `Caas Config`. The bugfix now handles all Exceptions, returning a HTTP 200 OK with the correct GraphQL error message.

[CMS-21011]

Bugfix: Extending a GraphQL Enum causes a ClassCastException

Extending a GraphQL Enum caused a ClassCastException while server start. This bug was fixed.

[CMS-20600]

Miscellaneous

Fixed a bug preventing apps to shutdown properly

Apps using the CoreMedia cache sometimes failed to destroy the cache instance leaving a thread named "coremedia-cache-CacheTimer" behind. This thread is now terminated when the spring application context shuts down.

[CMS-21009]

Third-Party Update: Tomcat

Tomcat has been updated to version 9.0.58 to avoid security vulnerabilities of the previous version.

[CMS-20961]

Third-Party Update: Spring Boot 2.5.8

In order to benefit from the improvements of the latest version of the Spring framework and to prepare for making use of the new features various 3rd party libraries have been updated.

- Caffeine 2.9.3
- GSON 2.8.9
- Hibernate 5.4.33
- HttpCore 4.4.15
- Micrometer 1.7.7
- Spring Boot 2.5.8
- Spring Data 2021.0.7
- Spring Framework 5.3.14
- Spring Security 5.5.4
- XmlUnit2 2.8.4

[CMS-20686]

Updated Protocol Buffers for Java to 3.19.3

Updated `com.google.protobuf:protobuf-java` dependencies to version 3.19.3 to fix known security vulnerabilities.

[CMS-20685]

Third-Party Update : Jackson

Jackson has been updated to version 2.12.6 to avoid security vulnerabilities of the previous version.

[CMS-20646]

Shared HCL/WCS Commerce Proxy enhancements

The blueprint based `commerce-proxy` in the Docker deployment was enhanced to better support shared HCL/WCS Commerce setups where multiple CMS systems share a single commerce system. Product Asset URLs using the `catalogimage` path are now postprocessed in the `commerce proxy` and the hostname is now correctly set to the proxied CMS hostname instead of the default `cmsHost` that is configured in the commerce system.

[CMS-20592]

Root Category Preview

We fixed the preview for the virtual root categories that do not have a corresponding shop URL. Instead of the previously rendered error, we now render its placements as a content page through the storefront. The layout is not accurate but it should give editorial users enough preview possibilities to create placement content that is used for inheritance to other subcategories.

[CMS-20430]

Images: Parameters in MIME types fixed

During transformation a lookup by MIME type must be done to find MIME type specific implementations. The MIME type was compared with parameters. MIME types are syntactically allowed to have parameters [see RFC 2046] but for images no parameters are specified. Unfortunately if there were parameters no specific implementation could be found even if it was possible to transform the image.

Now the MIME type parameters are ignored for the lookup.

[CMS-20198]

Fixed a bug preventing replacement of CAE richtext filter beans

`BlueprintRichtextFiltersConfiguration` does no longer expose richtext filter beans by their implementation types. It no longer references the richtext filter beans by type. It uses the filter bean names as qualifiers instead.

[CMS-19471]

Calista and Aurora UK site removed

The English / United Kingdom ("en_UK") demo content for Calista and Aurora Augmentation has been removed. The reason is that a default HCL Commerce system does not provide "en_UK" out of the box. When using the UK sites in Studio, the preview always showed the "en_US" storefront and pulled fragments from the "en_US" site. Any content changes made in the "en_UK" site were not reflected in the preview and this confused editorial users in a demo or testing scenario. This removal only affects demo content.

[CMS-18634]

4.5 Release 2110.1

Highlights in this Release

- The Workflow App, which opens fullscreen in a new browser tab, and in the new Studio design.
- New, exciting visual changes, illustrations, and animations, for example a login screen that comes in different variations based on your individual location and preferences.
- The new Preferred Site Selector, which enables an improved and intuitive locale and name sorting.
- Redesigned Toasts and Notifications, that minimize distractions and help focusing on important messages.
- The Content Issue Search, to tidy up your content repository, for example by increasing issue visibility through custom dashboard widgets in Studio.
- CKEditor 5 based Rich Text editing is available as a developer preview.
- Studio client is implemented in TypeScript instead of ActionScript/MXML
- Introduction of new Commerce Hub/Adapter API v2
- Upgrade to Spring Boot 2.5



The release contains:

- Breaking changes: 157
- Changes and improvements: 266
- Bugfixes: 365

Only issues which contain release notes are shown in this document. So, the numbers in the list above can be higher than the number of issues presented in the subsections.

Breaking Changes

- CMS-20068: **Removal: Deprecated Version of XliffExporter**
- CMS-20067: **Removal: Deprecated two-argument constructor of DefaultContentToTranslateItemTransformer**
- CMS-20066: **Removal: Deprecated Interface Method ContentObject-SiteAspect.getIgnoreUpdates**

- CMS-20039: Third-Party Update: Apache Tika and Transitive Dependencies
- CMS-20025: Removal: Deprecated `BaseCommandLineClient#addOptionValidated Method`
- CMS-20003: Removal: Deprecated `Site#getManagerGroupName Method`
- CMS-19998: Removal: Deprecated Options of Some Commandline Tools
- CMS-19996: Removal: Deprecated `AbstractVersionSelector#addOption Method`
- CMS-19993: Removal: Deprecated `CommandLineParameters#addOptionValidated Method`
- CMS-19992: Removal: Deprecated `AbstractUAPIClient#enableOutVerbose Method`
- CMS-19990: Removal: Deprecated `MarkupUtil#isEmptyMarkup Method`
- CMS-19768: New Feature: Workflow Final Actions and Archiving of Aborted Workflows
- CMS-19757: MongoDB Update to version 5.0.2
- CMS-19748: MediaStore Changes
- CMS-19606: Issue Search Filter Enabled By Default and Solr Schema Changes
- CMS-19599: Solr Base Image added
- CMS-19302: Changed Solr Terminology: Leader/Follower replaces Master/Slave
- CMS-19266: Third-Party Update: Apache Solr 8.8.2
- CMS-19112: Content Server and Workflow Server do not use `corem.home` anymore
- CMS-18965: Commerce Validator Issue Feeding
- CMS-17959: Removal: Deprecated Search and Feeder Configuration Properties
- CMS-16722: Improved CAE Feeder Database Usage
- CMS-7318: Removal: Deprecated Watchdog and Probedog infrastructure
- CMS-19187: Improved Translation Workflow Auto-Merge
- CMS-7021: Secure Blob Access
- CMS-2487: `blueprint-doctypes-xmlrepo.xml`: Available in Blueprint Sources
- CMS-20438: Removal: CAE Module `cap-editing-pbe-taglib-component`
- CMS-20436: Moved static CAE resources to `'/static/'`
- CMS-20342: Deprecation of Image Operations
- CMS-20323: Removal: CAE Code Minification
- CMS-20295: Removal: Elastic Social Personalization Extension
- CMS-20143: Deprecation: Property `"ieExpression"` of `CMAbstractCode`
- CMS-20048: Removal: Deprecated Sitemap URI Pattern `/service/sitemap/{sitemapPath}`

- CMS-20001: Removal: Deprecated UriComponentsHelper#addQueryParameters Method
- CMS-19984: Deprecation: ContentBlobHandlerBase
- CMS-19983: Removal: Deprecated ContentBlobHandlerBase#URIVARIABLE_VERSION Field
- CMS-19958: Removal: BlueprintFreemarkerFacade functions
- CMS-19409: Removal: Deprecated Methods of CMProductList
- CMS-19163: Removal: CAE TokenResolver Class
- CMS-19104: CAE cache-control default behavior changed
- CMS-19089: Removal: LiveContext IBM/WCS implementation
- CMS-18333: Moved internal CAE controllers to management application context
- CMS-15880: Removal: BlueprintFreemarkerFacade functions
- CMS-19278: Optimize Rendering of embedded content in Richtext
- CMS-19247: Removal: Sitemaps generation for commerce sites
- CMS-18512: Removal: CMHasContextsImpl#getDataViewFactory() method
- CMS-13015: Richtext Pagination
- CMS-19921: Removal: Deprecated TabManager Classes/Interfaces
- CMS-19750: Removal: CommerceConnection#getVendorXXX Methods
- CMS-19698: Removal: Studio Packages Proxy App
- CMS-19669: For contents having a "data"-property, the links for those blobs are now rendered instead of simple content links
- CMS-19295: Removal: Legacy Search Facets API from Studio
- CMS-18571: Moved: Modules Moved to new Package
- CMS-18021: Deprecation: Configuration hideIssuesButton in Control Room
- CMS-20399: BooleanPropertyField Label Changes
- CMS-19108: ILinkListWrapper Manipulation Methods Are Now Asynchronous By Signature
- CMS-18849: Fixed Inconsistent Usage of ILinkListWrapper#acceptsLinks
- CMS-16977: Simplified Most of the Configuration to [ExtConfig]
- CMS-19563: Redesign of Jobs, Toasts and Notifications
- CMS-19094: Added Search Filters for Library Extension
- CMS-18931: Validators
- CMS-20303: CatalogAlias configuration for single catalog setups
- CMS-19920: Use stable IDs in CAE Fragment Requests, if Possible

- CMS-19418: Optimized Taxonomy Performance
- CMS-19484: Banned Jakarta EE 9 Artifacts
- CMS-19116: Removal: Deprecated Properties
- CMS-18517: Updated to Spring 5.3.12 and Spring Boot 2.5.6
- CMS-17426: Removal: repository.workflow, repository.caplist, and usecaplist Properties
- CMS-20076: Switched to Sencha CMD 7.2.0.84
- CMS-19884: Mongo shell replacement mongosh
- CMS-16969: Renamed groupId com.coremedia.content-hub to com.coremedia.cms
- CMS-20138: Removal: IE11 Support in Themes and Bricks
- CMS-19955: Changed Package Manager in Frontend Workspace from yarn to pnpm
- CMS-19245: Third-Party Update: Node.js 16
- CMS-20063: Update to Latest Commerce Adapter Versions and Adaption of Configurations
- CMS-19923: Enable Readiness and Liveness Health Group Endpoints
- CMS-19249: Enables deployment of SFMC via docker-compose.
- CMS-19157: Removal: Deployment Archive with Chef Deployment
- CMS-18332: Container Image Build Process Changes
- CMS-16416: Traefik update in Docker Setup
- CMS-20308: Removal: Special Commerce Hub API for Preview URLs
- CMS-20130: Updated Multi Catalog Configuration
- CMS-20114: Removal: Deprecated Methods and Implementation
- CMS-19704: Removal: LC3 SearchService
- CMS-19695: Removal: Deprecated Developer Feature Candy Shop URLs
- CMS-19451: CurrentStoreContext and UserStoreContext API change
- CMS-19345: Removal: Obsolete Base Classes from LiveContext Test Modules
- CMS-19343: Removal: Hoverfly Extension Classes from LiveContext Test Modules
- CMS-19290: Removal: Deprecated Legacy Search API from LiveContext Extensions
- CMS-19221: Removal: Deprecations from LiveContext API
- CMS-19215: Removal: Deprecated Ic-extension Related LinkSchemes
- CMS-19214: Removal: com.coremedia.blueprint.base.livecontext.service Package
- CMS-19213: Removal: Deprecated Livecontext Classes and Methods
- CMS-19212: Removal: Deprecated Commerce Cache Invalidation Implementation
- CMS-19124: Removal: The ForVendor-Methods from the CommerceConnection

- CMS-17159: Simplified Registration of Custom Commerce Bean Types
- CMS-15229: Removal: CommerceConnection#getClone Method
- CMS-20151: GraphQL Java Libraries Updated to Version 17.3
- CMS-19836: Swagger Updated to Latest Version
- CMS-19796: Refactoring of Adapters
- CMS-19769: Filter Predicates May Now Access the DataFetchingEnvironment
- CMS-19685: Headless Server Deprecations for CoreMedia CMCC 11
- CMS-19641: New Feature: Plugin Extension Point FilterPredicate
- CMS-19428: AbstractDynamicListAdapter#getItems and SearchServiceAdapter#search now return DataFetcherResults
- CMS-19322: Removal: Deprecated Headless API for Commerce Entities
- CMS-19155: Removal of ViewBySiteDataFetchingEnvironment, Public API Changes of RemoteLinkWiringFactory and RemoteLinkDataFetcher
- CMS-19107: Addition: New RichText Type to Encapsulate the Different Representations of Richtext
- CMS-19076: Changed Configuration Properties
- CMS-19071: Ordered Content Schema
- CMS-18930: Site Query by Root Segment
- CMS-18315: Loading of Persisted Queries from Plugins
- CMS-17298: Content Types in Search are Now Validated
- CMS-20193: CMProductTeaser change in headless
- CMS-20085: Headless Commerce: Remove Autocompletion for Breadcrumb Parameter
- CMS-20014: Introduce a new field keywordsList which returns a list of keywords
- CMS-19941: JsonPreviewClient was refactored to Spring AutoConfiguration
- CMS-19885: Removal: ViewMapperConfig Class
- CMS-19452: Aligned Behavior of StructAdapter to Same as in SettingsAdapter
- CMS-19210: Default Java heap for Headless Server reset to 1 GB in Dockerfile
- CMS-19096: Headless Server Schema Changes
- CMS-19055: Support ContentBlobLinkComposer in RichText conversion
- CMS-20332: The site-manager container image is now also build with google jib
- CMS-20120: Support for IBM DB2 has been removed
- CMS-20049: Removal: Deprecated Private API BlobHelper Class
- CMS-20047: Removal: Deprecated Editorial Comments cache properties editorial.comments.data.cache.* Properties

- CMS-19999: Removal: Deprecated `SettingsPageResourceBundleFactory` Class
- CMS-19995: Removal: Deprecated `ThemeImporterImpl#legacyExternalLinkToDocumentName` Method
- CMS-19988: Removal: Deprecated `UserProvider2` Interface
- CMS-19987: Removal: Deprecated Legacy RuleProviders Classes
- CMS-19986: Removal: Deprecated `LdapMember#getOrganizationalUnit` Method
- CMS-19962: Removal: `management.health.contentRepository.enabled` Property
- CMS-19911: Migrated Blueprint Content Hub Adapters to Plugins
- CMS-19867: Dropped supported deprecated databases, servers, and browsers
- CMS-19858: New API For Custom Workflow Extension Point
- CMS-19851: Removal: Corporate Extension Modules for Content Server
- CMS-19850: Removal: Unused class `SiteToStoreContextCacheKeyWithTimeout`
- CMS-19816: New API for Content Type Localization
- CMS-19396: Removal: Old live context settings struct
- CMS-19254: Removal: Deprecated API of `TransformImageService`
- CMS-19224: Logging Standardization
- CMS-19060: Removal: Optimizely Extension
- CMS-18934: New Blueprint Parent
- CMS-18795: Replaced deprecated `org.junit.Assert.assertThat`
- CMS-18659: Replaced Guava Function and co. with JDK counterparts
- CMS-18658: Removal: Deprecated Consumer, Function, Predicate Classes
- CMS-17548: Implementations of Comparable: Added Type Arguments
- CMS-17559: Consolidate Blueprint groupIds and versions

Changes and Improvements

- CMS-20233: Deprecation: Obsolete classes in Workflow Java API
- CMS-19831: Added parametric HTTP endpoint to re-indexing CAE Feeder
- CMS-19758: Update of MongoDB Java Driver to version 4.3.0
- CMS-19502: Deprecation: `ListDiff#elementEqual`
- CMS-19378: Removal: Deprecation for `SiteModel`'s `siteIndicatorDepth` in Multi-Site
- CMS-19202: Third-Party Update: Apache Tika and Transitive Dependencies
- CMS-18880: `cm dump`: Dump Text Blobs to Console
- CMS-18545: Added endpoint to disable/enable the replicator service

- CMS-17957: Changed Content Feeder Configuration Properties
- CMS-17851: Enhanced Encryption Algorithm For System Passwords
- CMS-16834: Removal: `com.coremedia.elastic.core.test.Injection` and `core-test` module
- CMS-16043: New Validator: `PossiblyMissingMasterReferenceValidator` for Multi-Site
- CMS-11136: `validate-multisite` Improvement
- CMS-19719: Sitemap: Reduce Downtime by Improving the Recreation Strategy.
- CMS-19195: Cache-Control Strategy can use Full Request Paths
- CMS-18954: Configure CORS Allowed Origins with Studio Host for Preview CAE
- CMS-17119: It is now possible to run the CAE without setting a context path
- CMS-16628: Fixed handling of cyclic navigation structures in CAE
- CMS-11194: Deprecated Header and Footer in `CMChannel`
- CMS-20293: Fixed Quick Create Dialog Positioning
- CMS-20162: User Manager Full Text Search
- CMS-19887: Site Selector - Sorting by Locale
- CMS-19657: Update to CKEditor 4.16.1
- CMS-19596: Added Support for Empty Quick Create Name
- CMS-19347: Added Support for Multiple External Previews
- CMS-19149: Restored "Open In Tab" Checkbox for Bulk Uploads
- CMS-19049: Studio Client Now Has Referrer-Policy "same-site"
- CMS-18939: Start Publication Workflow Button From ActionsToolbar No Longer Includes "My Edited Contents" By Default
- CMS-19942: Multi-Selection Support for Taxonomy Link Lists
- CMS-19492: iFrame Support for Content Hub Preview
- CMS-19486: Disabling Notifications
- CMS-19039: Multiple Filter Support for Product List Documents
- CMS-18985: Configure Studio Forms
- CMS-18855: Fixed Error thrown for Empty UserManager Properties
- CMS-18055: Filtering Issues Along with Categories in Studio Library
- CMS-16100: Added Thumbnail Support for Content Hub
- CMS-15893: Added metrics to reflect the number of CMS user sessions
- CMS-20026: Update: JUnit 5, Mockito, PowerMock
- CMS-19388: Prevented creation of empty JAR files

- CMS-19886: New Supported File Types for Theme Importer
- CMS-19736: Added Lazy Loading to Images
- CMS-18953: Allow Standalone JavaScripts Files in Themes
- CMS-20127: Enabled Health Checks for Commerce Adapters
- CMS-20125: The wait-for-* scripts have been replaced with an environment variable controlled mechanism
- CMS-14653: Graceful Shutdown for all Spring Boot based Applications
- CMS-20183: Removal: Deprecated Properties
- CMS-19482: Commerce Hub Invalidation Events Added
- CMS-19420: Added Configurable Custom Attributes to Commerce Hub Commerce Beans
- CMS-19058: Catalog Alias Mapping Accepts Catalog ID
- CMS-20361: New ExtensionPoint PluginRestController
- CMS-20274: Add plugin extension point for custom scalar types
- CMS-19959: New Config Option: caas.stax.maxAliasesForCollections
- CMS-19926: New Extension Point for Plugins: WiringFactory
- CMS-19859: Add fullyQualifiedUrl field to CMDownload type
- CMS-19832: New Fields for Localized Content
- CMS-19792: New Plugin Extension Point: PluginSchemaAdapterFactory
- CMS-19643: GraphQL Java Library updated from 16.1 to Version 16.2
- CMS-19572: Adds New Endpoint to Support Filename and Extension in Media URL
- CMS-19480: Provide Generic Field for Retrieving in Linked Contents in Structs or Settings
- CMS-19468: Additional Headless Augmentation Calls without Commerce Connection
- CMS-19193: Query for Taxonomies by path
- CMS-18889: GraphQL Type PageGrid now Offers Name Property for Page Grid
- CMS-18878: MediaController Supports Conversion of Image Formats
- CMS-18844: New Fields Contain Lists of Content Objects Embedded in Richtexts
- CMS-18826: Custom Filter Queries
- CMS-18486: Loading of JSLT and REST Mappings From Plugins
- CMS-17909: Pagination for collections
- CMS-17512: New Plugin Extension Point: CopyToContextParameter
- CMS-20097: Configurable Log Pattern in Solr Base Image

- CMS-19985: Undeprecated CMLinkable#LOCAL_SETTINGS
- CMS-19953: Tomcat TLD Scanning has been Disabled for all Applications Except CAE
- CMS-19900: Added Headless Rest Mappings for Augmentation Queries
- CMS-19802: Third-Party Update: Apache Solr and Transitive Dependencies
- CMS-19734: Update of Commerce Adapter Versions
- CMS-19524: Third-Party Update: Spring Boot and Related Dependencies
- CMS-19456: Added the possibility to change the commerce reference of an augmented content
- CMS-19408: Query for Documents Tagged by Specific Taxonomies
- CMS-19368: Explode Spring Boot Applications for Commerce Adapters at Startup
- CMS-19166: Reproducible Maven Builds
- CMS-19118: Reworked Commerce search API
- CMS-18876: Removed Classic Studio Components
- CMS-18735: Added EntityController and StudioRestController as ExtensionPoints for Studio-Server Plugins
- CMS-17506: Unify Link Building for Commerce-Led and Content-Led URLs
- CMS-16276: Customization of CMQueryListImpl and SettingsStructToSearchQueryConverter

Bugfixes

- CMS-20313: Fixed Bug in LocalAndLinkedSettingsFinder
- CMS-19862: Update of coremedia-application-maven-plugin to 3.0.1
- CMS-19702: Third-Party Update: Spring-Security
- CMS-19642: Fixed CAE Feeder deadlock for JMX access
- CMS-19638: Third-Party Update: Apache PDFBox
- CMS-19636: CAE Feeder Performance Improvement
- CMS-19531: Fixed Output for cm encryptpasswordproperty
- CMS-19293: The 'property' column of the 'System' table has now the length 100
- CMS-19275: Verbosity of publishall
- CMS-19272: cm validate-multisite: Enhanced Robustness for Huge Multi-Site Setups
- CMS-19160: Blobs in structs get collected after version or content is destroyed
- CMS-19103: Fix import-user script if no user xml files found
- CMS-18680: Fixed Broken Order of Transformations

- CMS-18530: Fixed Solr hostname and port disclosure with malicious REST request
- CMS-18237: Improve Studio robustness when no WorkflowRepository is available
- CMS-18163: Fixed multi-threading issues in cleanrecyclebin tool
- CMS-16741: Fixed a bug in ObservedPropertyService leading to seemingly missing augmentations
- CMS-11312: Encrypt Initial Passwords
- CMS-9058: Fixed Content Feeder Handling of Destroyed Derived Content
- CMS-19671: Removed com.coremedia.elastic.social.springsecurity from component scan
- CMS-19600: Cache Classes Listing in DataViewFactory MBean fixed
- CMS-19527: Avoid Cache Deadlock for TransformedBlobKey
- CMS-19416: PageGrid Issue on Category and Product Pages Fixed
- CMS-19244: Fixed cache size handling of the DataViewFactory
- CMS-18988: CapBlobHandlerTest doesn't fail even if mediatransform.xml is populated
- CMS-18000: Sanitize user-data before logging it
- CMS-17644: Added hint for TransformedBlobCache usage
- CMS-20345: Window Skin removal
- CMS-20245: Fix commerce library lazy loading
- CMS-20221: Fixed Error On Closing CMPicture Document
- CMS-19707: FolderChooserListViewBase#getAvailablePathsExpression is now null robust
- CMS-19611: Fixed Link List Suggestions for Site Specific Taxonomies
- CMS-19595: Added "defaultNameValueExpression" Parameter to QuickCreateMenuItem
- CMS-19545: Modification of Issue Filter UI selection
- CMS-19491: Added new config preferredSiteOnly for LinkListPropertyField
- CMS-19382: CategoryDocumentForm Expands when there is an Issue
- CMS-19267: Fixed Caching Issue in ViewTypeUtil
- CMS-19243: Cache Buster is once again only applied for GET requests
- CMS-19158: Fixed some timing problems in FolderChooserList
- CMS-19138: Possible Format-Loss For Richtext in Studio Fixed
- CMS-19133: Fixed type error when aborting a workflow
- CMS-19050: Studio Client Now Has X-Content-Type-Options "nosniff" Header
- CMS-18549: Fixed Timing Problem when Loading Site Specific Settings

- CMS-17407: Fixed error when trying to republish a withdrawn folder
- CMS-16881: Controlroom: Start translation workflow button in toolbar hidden when unusable
- CMS-16737: Fixed endless loop bug when opening the Studio library
- CMS-16061: Fixed console warnings about deprecated Ext.EventManager
- CMS-20021: Fixed Notification Menu Closing Bug
- CMS-19951: Mimetype of uploaded xpi files is now recognized as application/x-xpinstall
- CMS-19905: Improve 'mayPreview' of ConfigureDocumentTypePlugin
- CMS-19902: Fixed Caching Bug for View Types
- CMS-19733: Error for Editing History of Deleted User Fixed
- CMS-19715: Fixed Issue with Tab Expand Plugin
- CMS-19682: Fixed Error in Job Tooltip
- CMS-19589: Reject requests against Studio Server until the application is live and ready
- CMS-19582: Fixed initialization of Studio ObjectMapper
- CMS-19389: Third-Party Update: Liquibase
- CMS-19323: Fixed Tooltip of Notification Menu Button
- CMS-19280: Embedding a CMHTML document into the richtext editor no longer leads to errors
- CMS-19279: Theme Importer
- CMS-19119: Fix StackoverflowError in Studio Server when starting sync workflow
- CMS-19109: Fixed Referrer Invalidations
- CMS-19086: Enhanced Error Handling For Editorial Comments When Connected to Old Server
- CMS-18892: Studio Locale Cookie with SameSite Lax
- CMS-17791: Fixed VisibilityValidator bug that caused NoSuchPropertyDescriptorException
- CMS-16028: BlobUploadXmpDataInterceptor merging existing local settings now
- CMS-15263: Fixed Problem With Vanished Characters in Text Fields / Areas
- CMS-19773: RemoveLikesTask updates counter "author:number_of_likes"
- CMS-18425: SimpleLinkFormatter link building properties configurable and protocol default to HTTPS
- CMS-20015: Fix warning about spring.profiles at application startup
- CMS-18834: File logging can be disabled

- CMS-19939: Extend live context environment with fragment connector disabled
- CMS-19675: Fixed propagation of CatalogAlias to CommerceBeans' StoreContexts
- CMS-19453: Fixed delegation to built-in commerce services for custom commerce service implementations
- CMS-19324: Fixed a bug regarding `commerce.hub.cache` properties
- CMS-20408: Refactor PluginConfiguration
- CMS-20315: Fix rendering of richtext in settings
- CMS-20007: Fix behaviour of CMHTML data property
- CMS-19701: Fix various nullability related issues
- CMS-19690: [Headless] Exposing Missing SEO Properties
- CMS-19689: Fix broken handling of arrays in input values for headless persisted queries and rest endpoint
- CMS-19622: Ignore destroyed contents in LinkListAdapter
- CMS-19517: Fix ViewBySiteFilterDataFetcher not checking DataFetcherResults
- CMS-19277: Configuration of custom Solr fields
- CMS-19073: SchemaParser ignores type extensions for inherited types
- CMS-19046: Headless Server sends HTTP header Strict-Transport-Security by default
- CMS-19045: Slightly improved error message handling
- CMS-18874: Spring Boot Endpoint Documentation added
- CMS-20426: Fix Check for Personal Data Usage
- CMS-20244: Third-Party Update : Tomcat
- CMS-19843: Third-Party Update : Ant
- CMS-19841: Third-Party Update : Tomcat
- CMS-19840: Third-Party Update : Jetty
- CMS-19830: ContentHub: ContentWriteInterceptors are now correctly applied for subtypes
- CMS-19257: Third-Party Update: Netty
- CMS-19250: Third-Party Update: Jetty
- CMS-19077: Update of MongoDB Java Driver
- CMS-18890: Fixed OSM Marker Initialization

4.5.1 Breaking Changes

CoreMedia Core

Removal: Deprecated Version of XliffExporter

The deprecated interface `com.coremedia.translate.xliff.exporter.XliffExporter` [cap-xliff] and its related interfaces and classes have been removed.

Follow [Section "Removal: Deprecated Version of XliffExporter" \[187\]](#) for upgrade information.

[CMS-20068]

Removal: Deprecated two-argument constructor of DefaultContentToTranslateItemTransformer

The deprecated two-argument constructor of `com.coremedia.translate.item.DefaultContentToTranslateItemTransformer` [cap-translate-item] has been removed:

Follow [Section "Removal: Deprecated two-argument constructor of DefaultContentToTranslateItemTransformer" \[189\]](#) for upgrade information.

[CMS-20067]

Removal: Deprecated Interface Method ContentObjectSiteAspect.getIgnoreUpdates

The deprecated interface method `ContentObjectSiteAspect.getIgnoreUpdates()` has been removed.

Follow [Section "Removal: Deprecated Interface Method ContentObjectSiteAspect.getIgnoreUpdates" \[189\]](#) for upgrade information.

[CMS-20066]

Third-Party Update: Apache Tika and Transitive Dependencies

Apache Tika has been updated to new major version 2.1.0.

Follow [Section "Third-Party Update: Apache Tika and Transitive Dependencies" \[190\]](#) for upgrade information.

[CMS-20039]

Removal: Deprecated BaseCommandLineClient#addOptionValidated Method

The deprecated method `com.coremedia.cmdline.BaseCommandLineClient#addOptionValidated` has been removed from the Java API. If you still use it in your project code, switch to `options.addOption(option)`, `options` and `option` referring to the arguments of `addOptionValidated`.

[CMS-20025]

Removal: Deprecated Site#getManagerGroupName Method

The deprecated method `com.coremedia.cap.multisite.Site#getManagerGroupName` has been deleted. If you still use it in your project code, replace it with `getManagerGroupNames().stream().findFirst().orElse("")` -- mind, though, that it is better respecting all group names returned by `getManagerGroupNames()`.

[CMS-20003]

Removal: Deprecated Options of Some Commandline Tools

The command line tools `publish`, `destroy` and `query` do not support the option `paths` any longer. Use `path` instead (also CSV capable, despite the singular name). `paths` has already been undocumented for a while, so you might not even be aware that it existed.

[CMS-19998]

Removal: Deprecated AbstractVersionSelector#addOption Method

The deprecated method `com.coremedia.cmdline.management.AbstractVersionSelector#addOption` has been removed from the Java API. If you still use it in your project code, switch to `options.addOption(option)`, `options` and `option` referring to the arguments of `addOption`.

[CMS-19996]

Removal: Deprecated CommandLineParameters#addOptionValidated Method

The deprecated method `com.coremedia.cmdline.CommandLineParameters#addOptionValidated` has been removed from the Java API. If you still use it in your project code, switch to `options.addOption(option)`, `options` and `option` referring to the first two arguments of `addOptionValidated`.

[CMS-19993]

Removal: Deprecated AbstractUAPIClient#enableOutVerbose Method

The deprecated method `com.coremedia.cmdline.AbstractUAPIClient#enableOutVerbose` has been deleted from the Java API. If you are still using it in your project code, switch to `com.coremedia.cmdline.BaseCommandLineClient#enableVerboseLogging`.

[CMS-19992]

Removal: Deprecated MarkupUtil#isEmptyMarkup Method

`com.coremedia.xml.MarkupUtil#isEmptyMarkup` has been deleted. Use `#isEmptyRichtext` (for CoreMedia Richtext only) or `#hasText` (for markup of arbitrary grammars) instead.

[CMS-19990]

New Feature: Workflow Final Actions and Archiving of Aborted Workflows

Workflows can now declare custom final actions, which are executed after a workflow process has completed successfully or was aborted. To this end, a workflow definition can contain `<FinalAction>` elements with custom classes that implement the interface `com.coremedia.cap.workflow.plugin.FinalAction`.

Follow [Section "New Feature: Workflow Final Actions and Archiving of Aborted Workflows" \[193\]](#) for upgrade information.

[CMS-19768]

MongoDB Update to version 5.0.2

MongoDB has been updated to version 5.0.2 for the Docker Deployment. Note that an upgrade is only possible from a 4.4.4 version. An upgrade from an earlier version needs to be done successively. Please refer to the official MongoDB documentation on how to upgrade: <https://docs.mongodb.com/v5.0/release-notes/5.0-upgrade-standalone/>

[CMS-19757]

MediaStore Changes

Database

Follow [Section "MediaStore Changes" \[194\]](#) for upgrade information.

[CMS-19748]

Issue Search Filter Enabled By Default and Solr Schema Changes

The Solr index schema for *Content Feeder* and *CAE Feeder*, and the configuration of the *Content Feeder* was changed to enable the search filter for validation issues by default, and to also support Solr nested documents in the *CAE Feeder*.

Follow [Section "Issue Search Filter Enabled By Default and Solr Schema Changes" \[194\]](#) for upgrade information.

[CMS-19606]

Solr Base Image added

A new base image has been added for the Solr search engine. The image can be found on DockerHub at [coremedia/solr-base](#). The image adds the scripting layer to setup a leader / follower setup in the standalone Solr mode.

Follow [Section "Solr Base Image added" \[195\]](#) for upgrade information.

[CMS-19599]

Changed Solr Terminology: Leader/Follower replaces Master/Slave

Apache Solr has replaced the names of their replication setup from Master/Slave to Leader/Follower. These names have now been changed in Solr configuration files (`solrconfig.xml`) and CoreMedia documentation. For details of the change in Solr, see https://solr.apache.org/guide/8_8/solr-upgrade-notes.html#solr-8-7 .

Follow [Section “Changed Solr Terminology: Leader/Follower replaces Master/Slave” \[195\]](#) for upgrade information.

[CMS-19302]

Third-Party Update: Apache Solr 8.8.2

Apache Solr has been updated to version 8.8.2, which fixes some security vulnerabilities of the previous version.

Follow [Section “Third-Party Update: Apache Solr 8.8.2” \[196\]](#) for upgrade information.

[CMS-19266]

Content Server and Workflow Server do not use `corem.home` anymore

The `corem.home` directory and system property are no longer used by the Content Server and Workflow Server.

Follow [Section “Content Server and Workflow Server do not use `corem.home` anymore” \[197\]](#) for upgrade information.

[CMS-19112]

Commerce Validator Issue Feeding

Commerce related validator issues defined in `LcStudioValidatorsConfiguration` are now also feeded into the Solr index "studio" by default. Therefore the *Content Feeder* now also uses *Commerce Hub* connections to configured *Commerce Adapters* . The legacy (and deprecated) LiveContext Blueprint extension for IBM/HCL B2B is **not** supported.

Follow [Section “Commerce Validator Issue Feeding” \[198\]](#) for upgrade information.

[CMS-18965]

Removal: Deprecated Search and Feeder Configuration Properties

Several deprecated configuration properties have been removed for the *Content Feeder* , the *CAE Feeder* and for applications that connect to Solr for searching.

Follow [Section “Removal: Deprecated Search and Feeder Configuration Properties” \[199\]](#) for upgrade information.

[CMS-17959]

Improved CAE Feeder Database Usage

The *CAE Feeder* was improved to store data more efficiently in the database, which reduces database disk space requirements and improves throughput. To this end, the data type of some database columns was changed from a string type to a binary type. This change also applies to custom applications based on the *Proactive Engine*.

Follow [Section "Improved CAE Feeder Database Usage" \[200\]](#) for upgrade information.

[CMS-16722]

Removal: Deprecated Watchdog and Probedog infrastructure

The Watchdog and Probedog components have been removed. To implement the same functionality you can use Spring Boot actuators together with an orchestration solution. In addition to the standard actuators of Spring Boot, CoreMedia adds a set of custom endpoints to represent the health state of the applications. You find the documentation in the operations basics handbook.

Follow [Section "Removal: Deprecated Watchdog and Probedog infrastructure" \[200\]](#) for upgrade information.

[CMS-7318]

Improved Translation Workflow Auto-Merge

The "Auto-Merge" feature used in translation workflows to copy property values from master to derived content items has been improved. In previous releases, it completely ignored translatable properties and never changed such properties in the derived content. This has been changed because it could lead to unexpected results, especially for properties nested in structs. Furthermore, the merge algorithm was improved for link annotations.

Follow [Section "Improved Translation Workflow Auto-Merge" \[201\]](#) for upgrade information.

[CMS-19187]

Secure Blob Access

Blob access by clients is protected against URL guessing now. If you want to use components from earlier CMCC releases with new content servers, you must set the application property `cap.server.blob-md5-permission-check` to `false` for the new content servers.

[CMS-7021]

blueprint-doctypes-xmlrepo.xml: Available in Blueprint Sources

The `blueprint-doctypes-xmlrepo.xml` located in so-called Blueprint-Base, unavailable to most customers, now has a copy in the Blueprint module `test-util`.

Follow [Section “blueprint-doctypes-xmlrepo.xml: Available in Blueprint Sources” \[202\]](#) for upgrade information.

[CMS-2487]

CoreMedia Content Application Engine

Removal: CAE Module `cap-editing-pbe-taglib-component`

The unused and deprecated module `cap-editing-pbe-taglib-component` has been removed. It is recommended to use the tags `cm:previewScripts` and `cm:metadata` of the CAE tag library in JSP templates instead of the removed tags `pbe:head` and `pbe:contextinfo`, respectively.

[CMS-20438]

Moved static CAE resources to `'/static/'`

Build-in static CAE resources were moved to `'/static/'`. Although this it is not a hard requirement that static resources be served from the `'/static/'` path we do recommend to serve static resources using this prefix because it supports a clean separation of static resources from servlet mappings when developing rewrite rules.

[CMS-20436]

Deprecation of Image Operations

The image operation `gamma` and the parameters of the `unsharpMask` are now deprecated. Both operations couldn't be controlled by editors in the Studio directly for single images. They also haven't been used in the Blueprint by default.

Follow [Section “Deprecation of Image Operations” \[203\]](#) for upgrade information.

[CMS-20342]

Removal: CAE Code Minification

The deprecated and unused interface `com.coremedia.blueprint.cae.view.processing.Minifier` has been removed. It is recommended to minify and optimize CSS and JavaScript files during build steps in the frontend workspace. This will be done by the blueprint default webpack configuration. Therefore the property `disableServerSideCompression` has been removed from Theme Configuration and build scripts too.

[CMS-20323]

Removal: Elastic Social Personalization Extension

The Elastic Social Personalization extension was removed without replacement.

[CMS-20295]

Deprecation: Property "ieExpression" of CMAbstractCode

The doctype property `ieExpression` of `CMAbstractCode` has been deprecated due to removal of Internet Explorer support. All usages will be removed soon.

[CMS-20143]

Removal: Deprecated Sitemap URI Pattern `/service/sitemap/{sitemapPath}`

The deprecated Sitemap URI Pattern `/service/sitemap/{sitemapPath}` and the related handler have been removed from `com.coremedia.blueprint.cae.sitemap.SitemapHandler` [`cae-base-lib`]. The same applies to private members `ACTION_NAME` and `SITEMAP_PATH` which are now obsolete.

Follow [Section "Removal: Deprecated Sitemap URI Pattern `/service/sitemap/{sitemapPath}`" \[204\]](#) for upgrade information.

[CMS-20048]

Removal: Deprecated `UriComponentsHelper#addQueryParameters` Method

The deprecated method `com.coremedia.objectserver.web.links.UriComponentsHelper#addQueryParameters` has been deleted. If you still use it in your project code, switch to `UriComponentsBuilder#queryParams (MultiValueMap)`

[CMS-20001]

Deprecation: `ContentBlobHandlerBase`

`com.coremedia.objectserver.web.ContentBlobHandlerBase` has been deprecated. No handlers use it any longer. If you still need it in your project code, please contact the CoreMedia support.

[CMS-19984]

Removal: Deprecated `ContentBlobHandlerBase#URIVARIABLE_VERSION` Field

The deprecated field `com.coremedia.objectserver.web.ContentBlobHandlerBase#URIVARIABLE_VERSION` has been deleted from the Java API. The result map of `ContentBlobHandlerBase#buildBlobLink` contains no entry for this key any longer. If you still use it in your project, migrate to `ContentBlobHandlerBase#URIVARIABLE_ETAG`.

[CMS-19983]

Removal: `BlueprintFreemarkerFacade` functions

Both deprecated Freemarker functions `getContainerMetadata()` and `getDisplaySize()` have been removed. Please use `Container#getContainerMetadata()` and `getDisplayFileSize()` instead.

[CMS-19958]

Removal: Deprecated Methods of CMProductList

The deprecated Methods from the CMProductList interface have been removed. The implementation of those methods are still available in 'CMProductListImpl.java' as they are used in 'livecontext-dataviews.xml'.

[CMS-19409]

Removal: CAE TokenResolver Class

Classes `com.coremedia.objectserver.web.links.TokenResolver` and `com.coremedia.objectserver.web.links.TokenResolverHelper` were removed because token resolving repeatedly caused encoding issues with links. Use Spring utilities to build URL components and their built-in support to replace variables instead.

[CMS-19163]

CAE cache-control default behavior changed

In CAE responses the "s-maxage" value in Cache-Control header was forced to be set to the same value as for "max-age", regardless of whether there was a separate configuration for "s-maxage". A factor can now be configured for setting the "s-maxage" header value in relation to "max-age" [see new property "cae.cache-control.s-maxage-factor"]. Default value is "0.0" which means that no "s-maxage" is set unless it is explicitly configured. The old behavior can be restored by setting the value "1.0".

[CMS-19104]

Removal: LiveContext IBM/WCS implementation

The deprecated LiveContext HCL/IBM extension in the blueprint has been removed. It is replaced by the Commerce Hub. Connections to an eCommerce system are no longer established directly, but only via Commerce Hub/Adapter. The example Aurora B2B sites have been removed from the demo content. As a result, a few features of the B2B support have also been omitted. Note, there is still no replacement in the Commerce Hub for them. In detail, these are the handling of workspaces and contracts.

[CMS-19089]

Moved internal CAE controllers to management application context

In CM10 the CAE exposed several endpoints below `'/blueprint/servlet/internal'` which had to be handled separately when setting up URL rewrite rules so that they cannot be accessed from the public internet. In CM11 they are now configured to be included in the spring boot management context (together with the actuators) and are available on a different port (by default in blueprint deployment port 8081 on the CAE Docker Container and mapped to 40981 for preview and 42181 for live CAE). Affected endpoints are concerned with URL generation (`/blueprint/servlet/internal/service/url`), sitemap generation (`/blueprint/servlet/internal/<siteid>/sitemap-org`) and test URL generation (`/blueprint/servlet/internal/<siteid>/testurls`).

Follow [Section “Moved internal CAE controllers to management application context” \[207\]](#) for upgrade information.

[CMS-18333]

Removal: BlueprintFreemarkerFacade functions

Both deprecated Freemarker functions `createBeanFor()` and `createBeansFor()` have been moved from `BlueprintFreemarkerFacade` to `LivecontextFreemarkerFacade`, because of the remaining usages in livecontext templates. We recommend to not use the functions in projects.

[CMS-15880]

Optimize Rendering of embedded content in Richtext

When embedding content (esp. images) in richtext, `asRichtextEmbed` templates are likely to break HTML validity by rendering `<div>` blocks inside the richtext's `<p>` context. So far, the Blueprint's `PDivUntanglingFilter` tried to be smart and adjust the HTML accordingly by closing and reopening the `<p>` context around an embedded object. However, this magic never really matched projects' expectations about the result, because it sometimes led to unwanted additional paragraphs or empty links.

Follow [Section “Optimize Rendering of embedded content in Richtext” \[208\]](#) for upgrade information.

[CMS-19278]

Removal: Sitemaps generation for commerce sites

The sitemap generation for commerce sites is removed from the CAE. The implementation was part of the old IBM livecontext extension that was also removed. The feature can be re-implemented by integrating the `com.coremedia.livecontext.ecommerce.ibm.cae.sitemap` package from CMCC 10 blueprint.

[CMS-19247]

Removal: CMHasContextsImpl#getDataViewFactory() method

`CMHasContextsImpl#getDataViewFactory()` has been removed, since its usage could lead to cyclic dependencies or uncontrollable cache invalidations.

[CMS-18512]

Richtext Pagination

`ParagraphHelper#createParagraphs(Markup, int)` was supposed to paginate Richtext at headings [`<p>` elements with class attributes `p-\-heading-1-4`] ever since. Due to a bug, this feature was broken. Now, it is fixed, so you might observe more pages. This concerns especially `CMTeasableImpl#getTextAsParagraphs`. If you don't like the fixed pagination, omit the change in `StAXBlockElementPaginator.java:81` when you update the Blueprint.

[CMS-13015]

CoreMedia Studio Client

Removal: Deprecated TabManager Classes/Interfaces

The following deprecated classes/interfaces have been removed from `editor-components`:

Follow [Section "Removal: Deprecated TabManager Classes/Interfaces" \[209\]](#) for upgrade information.

[CMS-19921]

Removal: CommerceConnection#getVendorXXX Methods

The following methods of the interface `com.coremedia.livecontext.ecommerce.common.CommerceConnection` were removed without replacement:

Follow [Section "Removal: CommerceConnection#getVendorXXX Methods" \[210\]](#) for upgrade information.

[CMS-19750]

Removal: Studio Packages Proxy App

The Studio Packages Proxy App has been removed. The new plugin mechanism can be used instead. See documentation for details.

[CMS-19698]

For contents having a "data"-property, the links for those blobs are now rendered instead of simple content links

Render links for blobs in the headless richtext instead of the previous content links which weren't optimal for content types like CMDownload or CMPicture where you might just be interested in the link to the data.

[CMS-19669]

Removal: Legacy Search Facets API from Studio

The support for the legacy commerce search facets had been removed from the Studio client and backend. The facet editor for product lists does only support the multi-facet selection from now on.

Follow [Section "Removal: Legacy Search Facets API from Studio" \[210\]](#) for upgrade information.

[CMS-19295]

Moved: Modules Moved to new Package

The following modules have been moved to the new npm package `@coremedia/studio-client.main.ckeditor4-components`

Follow [Section “Moved: Modules Moved to new Package” \[211\]](#) for upgrade information.

[CMS-18571]

Deprecation: Configuration `hideIssuesButton` in Control Room

Deprecated configuration `hideIssuesButton` of `com.coremedia.cms.editor.controlroom.workflow.translation.DefaultTranslationWorkflowDetailForm` as it has no effect anymore. The property will be removed in the next major release

[CMS-18021]

BooleanPropertyField Label Changes

We fixed a bug where BooleanPropertyFields had no tooltip. For this, we changed an internal plugin of the field. To this end, resource bundles need to be adapted to display the correct field labels.

Follow [Section “BooleanPropertyField Label Changes” \[212\]](#) for upgrade information.

[CMS-20399]

ILinkListWrapper Manipulation Methods Are Now Asynchronous By Signature

We needed to change the behaviour of `#setLinks()`, `#addLinks()`, `#addLinksAtIndex()`, `#moveLinks()`, `#moveLinksToIndex()` and `#removeLinksAtIndex()` for some important implementations of `ILinkListWrapper` to be asynchronous. As a consequence, we changed the signature of these methods to return a Promise. In custom implementations where no asynchronicity is needed, the code can simply be changed as in the following example.

Follow [Section “ILinkListWrapper Manipulation Methods Are Now Asynchronous By Signature” \[212\]](#) for upgrade information.

[CMS-19108]

Fixed Inconsistent Usage of `ILinkListWrapper#acceptsLinks`

The function `ILinkListWrapper#acceptsLinks` had unclear documentation and therefore was neither used nor implemented consistently.

Follow [Section “Fixed Inconsistent Usage of `ILinkListWrapper#acceptsLinks`” \[213\]](#) for upgrade information.

[CMS-18849]

Simplified Most of the Configuration to `[ExtConfig]`

We simplified most of our configuration for Ext-Components from `Bindable` to `ExtConfig`. In case you are not working only with our Public API (and not, for example, with native

JavaScript in Studio) this has no impact. Otherwise you might need to change access to configuration in your native JavaScript from "getSomeConfig" and "setSomeConfig" to "someConfig".

[CMS-16977]

CoreMedia Studio Server

Redesign of Jobs, Toasts and Notifications

The running jobs window has been redesigned and aligned with the new Studio design.

Follow [Section "Redesign of Jobs, Toasts and Notifications" \[214\]](#) for upgrade information.

[CMS-19563]

Added Search Filters for Library Extension

The Studio library does support search filters for extensions now. As a result, search filters can now be added for commerce integration. The existing search filters work as before and saved searches aren't affected by this change.

Follow [Section "Added Search Filters for Library Extension" \[215\]](#) for upgrade information.

[CMS-19094]

Validators

You can declare validators not only as Spring beans, but alternatively also by JSON configuration files now. This is supported for most core validator classes, except of those which make only sense as singletons and which are provided by default, like the `AvailableLocalesValidator`. Most Blueprint validator classes can easily be enabled for Json configuration in the project, if needed.

Follow [Section "Validators" \[216\]](#) for upgrade information.

[CMS-18931]

CatalogAlias configuration for single catalog setups

Configured catalogAlias below `commerce#catalogConfig#catalogAlias` is used to store commerce ids from now on.

Follow [Section "CatalogAlias configuration for single catalog setups" \[216\]](#) for upgrade information.

[CMS-20303]

Use stable IDs in CAE Fragment Requests, if Possible

So far the id parameter of commerce fragment requests from the CoreMedia fragment connector to the CAE were treated as external technical IDs, whereas all other commu-

nication uses stable external IDs for commerce entities. This leads to unnecessary cache misses, commerce communication and memory consumption in the CAE and commerce adapter caches.

Follow [Section “Use stable IDs in CAE Fragment Requests, if Possible” \[217\]](#) for upgrade information.

[CMS-19920]

Optimized Taxonomy Performance

The performance of the *DefaultTaxonomy* has been improved. This has been achieved through better UAPI caching. In addition, the property *maxDocumentsPerFolder* can be set now. This property will trigger the creation of new taxonomy documents created by the Taxonomy Manager in a new sub-folder named with a UUID once the default folder has reached the configured amount of documents. New sub-folders are created this way every time when the last used one has reached its limit.

Follow [Section “Optimized Taxonomy Performance” \[217\]](#) for upgrade information.

[CMS-19418]

CoreMedia Workspace

Banned Jakarta EE 9 Artifacts

There is a mismatch between Maven coordinates and Java packages of Jakarta EE 8/9 artifacts. [For details see https://www.eclipse.org/community/eclipse_newsletter/2020/november/1.php .] We follow the strategy of Spring Boot 2.5 of how to deal with this problem and ban all Jakarta EE 9 artifacts. If you use Jakarta EE 9 artifacts in your project, please try to replace them with the according Jakarta EE 8 libraries.

[CMS-19484]

Removal: Deprecated Properties

Properties that were deprecated because they were replaced by Spring Boot configuration properties have now been removed:

Follow [Section “Removal: Deprecated Properties” \[218\]](#) for upgrade information.

[CMS-19116]

Updated to Spring 5.3.12 and Spring Boot 2.5.6

In order to benefit from the improvements of the latest version of the Spring framework and to prepare for making use of the new features various 3rd party libraries have been updated.

Follow [Section “Updated to Spring 5.3.12 and Spring Boot 2.5.6” \[225\]](#) for upgrade information.

[CMS-18517]

Removal: repository.workflow, repository.capikey, and useapikey Properties

Removed the properties `repository.workflow`, `repository.capikey`, and `useapikey`. Use `repository.workflow.connect` instead of `repository.workflow` and `repository.capikey.connect` instead of `repository.capikey` and `useapikey`.

[CMS-17426]

Switched to Sencha CMD 7.2.0.84

Please use Sencha CMD 7.2.0.84 to build the studio-client to avoid heap space related problems after switching to the new NPM/TypeScript based tooling.

[CMS-20076]

Mongo shell replacement mongosh

The mongo shell has been deprecated with MongoDB 5.0. The replacement shell is mongosh. Usages of `mongo` shell have been replaced with `mongosh` in documentation, deployment scripts and the example script `dump-es-user-data.js`.

[CMS-19884]

Renamed groupId com.coremedia.content-hub to com.coremedia.cms

The groupId of the content-hub core artifacts has been changed from `com.coremedia.content-hub` to `com.coremedia.cms` in order to adjust to naming conventions.

[CMS-16969]

CoreMedia Frontend Workspace

Removal: IE11 Support in Themes and Bricks

The example themes and bricks in the frontend workspace do not support Internet Explorer 11 anymore by removing it from the `browserlist` entry in the `package.json`. Also extra polyfills for IE11 in the blueprint bricks and in the tooling have been removed. You can still add polyfills or custom styles for IE11 to themes and bricks in your projects. You may need to adapt the webpack and babel configuration.

[CMS-20138]

Changed Package Manager in Frontend Workspace from yarn to pnpm

The package manager of the frontend workspace has been changed from `yarn` to `pnpm`. `Yarn` in version 1 is not maintained anymore and `pnpm` is fast, efficient, strict and supports monorepos without additional libraries like `lerna`. Please check <https://pnpm.io/> for more details and installation. Please use the latest 6.x version of `pnpm`. Changes to the tooling should not affect any bricks or themes in your project.

Follow [Section “Changed Package Manager in Frontend Workspace from yarn to pnpm” \[227\]](#) for upgrade information.

[CMS-19955]

Third-Party Update: Node.js 16

Node.js has been updated to the latest LTS version 16. Please update your local installation to version 16 for local development too. Node.js version 14 is not supported!

[CMS-19245]

CoreMedia Deployment

Update to Latest Commerce Adapter Versions and Adaption of Configurations

The [default] versions of the commerce adapters have been updated to the latest 2.0.x versions. The new adapter versions require some configuration changes:

Follow [Section “Update to Latest Commerce Adapter Versions and Adaption of Configurations” \[228\]](#) for upgrade information.

[CMS-20063]

Enable Readiness and Liveness Health Group Endpoints

With Spring Boot 2.3 it is possible to define two extra subpaths of the health actuator to be used as the probing endpoints of the liveness and readiness checks in Kubernetes. This change enables those groups and adds the relevant checks to these groups.

Follow [Section “Enable Readiness and Liveness Health Group Endpoints” \[229\]](#) for upgrade information.

[CMS-19923]

Enables deployment of SFMC via docker-compose.

Includes `global/deployment/docker/compose/sfmc.yml` to deploy SFMC via docker-compose.

[CMS-19249]

Removal: Deployment Archive with Chef Deployment

The Deployment Archive has been removed. Please use the container images build by the `*-app` Maven modules to deploy CoreMedia. The Docker compose development setup, shows what images need to be used, what environment variables are mandatory and how ports and volumes need to be mapped.

[CMS-19157]

Container Image Build Process Changes

The `dockerfile-maven-plugin` has been replaced with the `google-jib-plugin` to build OCI conform container images, because the `dockerfile-maven-plugin` is no longer maintained. The `google-jib-plugin` integrates well with Spring-Boot applications and allows to build images securely without Docker being installed. This allows for a rootless, daemonless build but restricts the image building process to use only adding image layers by adding resources. There is no possibility to execute any `RUN` steps during the build process. Building the images without docker now requires a configured container registry although it is still possible to build images in the local docker daemon, without the need of a registry.

Follow [Section "Container Image Build Process Changes" \[230\]](#) for upgrade information.

[CMS-18332]

Traefik update in Docker Setup

Traefik has been updated to 2.5.

Follow [Section "Traefik update in Docker Setup" \[232\]](#) for upgrade information.

[CMS-16416]

CoreMedia Commerce Hub

Removal: Special Commerce Hub API for Preview URLs

The special treatment of Studio preview urls has been removed from Commerce Hub API. Now preview urls are generated the same way as the other commerce related links via LinkTemplates and dynamic replacement values. Most of the changes took place within the commerce adapter workspaces. On the CMS side the `PreviewUrlService` has been deleted and there are changes in the `CommerceStudioLinks`.

Follow [Section "Removal: Special Commerce Hub API for Preview URLs" \[232\]](#) for upgrade information.

[CMS-20308]

Updated Multi Catalog Configuration

The catalog alias configuration for the multi catalog feature has been updated. The former Studio content configuration setting "livecontext.catalogAliases" has been removed and needs to be updated, if you use multiple catalogs for one site. If you do not use the multi catalog feature there is no action required.

Follow [Section "Updated Multi Catalog Configuration" \[233\]](#) for upgrade information.

[CMS-20130]

Removal: Deprecated Methods and Implementation

Removed deprecated methods `getCapacity/getCacheTimesInSeconds` of class `com.coremedia.blueprint.base.livecontext.ecommerce.common.CommerceCache`. Also removed its `isEnabled` method because the commerce cache can no longer be disabled.

Follow [Section "Removal: Deprecated Methods and Implementation" \[234\]](#) for upgrade information.

[CMS-20114]

Removal: LC3 SearchService

Removed the interface `com.coremedia.livecontext.ecommerce.search.SearchService` because commerce hub does not support search suggestions. Search is supported via `com.coremedia.livecontext.ecommerce.catalog.CatalogService#search`.

[CMS-19704]

Removal: Deprecated Developer Feature Candy Shop URLs

The deprecated developer feature "Candy Shop URLs" has been removed from the Blueprint. There is no replacement planned, developers will need to configure their local machine in the shop configuration to retrieve fragments from their localhost. Frontend development is easily possible through the Frontend Developer workflow and uploading of local Themes to the CMS.

[CMS-19695]

CurrentStoreContext and UserStoreContext API change

Deprecated methods have been removed from `com.coremedia.blueprint.base.livecontext.ecommerce.common.CurrentStoreContext` and `com.coremedia.blueprint.base.livecontext.ecommerce.common.CurrentUserContext`.

Follow [Section "CurrentStoreContext and UserStoreContext API change" \[234\]](#) for upgrade information.

[CMS-19451]

Removal: Obsolete Base Classes from LiveContext Test Modules

With the introduction of the Commerce Hub, the connections to commerce systems have been moved away from the central CMS services to the commerce adapters. After all legacy LiveContext extensions (those, that don't use the commerce adapters) have been removed, their base test classes are now obsolete and have also been removed.

[CMS-19345]

Removal: Hoverfly Extension Classes from LiveContext Test Modules

With the introduction of the Commerce Hub, the connections to commerce systems have been moved away from the central CMS services to the commerce adapters. The tests that require commerce connections (whose requests and responses can be recorded and replayed with Hoverfly) have been moved, too, making the Hoverfly extension classes in LiveContext test modules obsolete, hence they have now been removed.

[CMS-19343]

Removal: Deprecated Legacy Search API from LiveContext Extensions

CMCC11 doesn't support old adapter versions with legacy search API anymore. Legacy commerce client API and implementations have been removed. This includes:

Follow [Section "Removal: Deprecated Legacy Search API from LiveContext Extensions" \[236\]](#) for upgrade information.

[CMS-19290]

Removal: Deprecations from LiveContext API

The old LiveContext API is used to access the Commerce Hub / generic commerce client from Studio and other clients. Some deprecations have now been deleted that are not needed anymore or that are not supported by the Commerce Hub. They are in detail: the AvailabilityService (not supported by the CommerceHub, should be requested directly from the commerce system), UserServer and UserSessionService (not needed anymore, the whole part of session sync between the CMS and the commerce system is gone, only the login/logout can be done via the commerce adapter), the AssetUrlProvider (is also not needed anymore).

[CMS-19221]

Removal: Deprecated lc-extension Related LinkSchemes

Obsolete deprecated linkschemes used for commerce integration via lc-extension have been deleted.

[CMS-19215]

Removal: com.coremedia.blueprint.base.livecontext.service Package

The deprecated `com.coremedia.blueprint.base.livecontext.service` package has been removed without replacement as it was part of the legacy

Blueprint commerce integration. The Commerce Hub integration does not yet offer corresponding classes dealing with connections to Storefront services.

Follow [Section “Removal: `com.coremedia.blueprint.base.livecontext.service` Package” \[237\]](#) for upgrade information.

[CMS-19214]

Removal: Deprecated Livecontext Classes and Methods

The classes `AbstractCommerceBean` and `SpringCommerceBeanFactory` from package `com.coremedia.blueprint.base.livecontext.ecommerce.common` were removed. Use `com.coremedia.blueprint.base.livecontext.client.beans.ClientCommerceBeanFactoryMethodRegistry` instead.

Follow [Section “Removal: Deprecated Livecontext Classes and Methods” \[237\]](#) for upgrade information.

[CMS-19213]

Removal: Deprecated Commerce Cache Invalidation Implementation

The deprecated source code for the Commerce Cache Invalidation of the LiveContext 3 extensions has been removed. Customizations based on that implementation have to be updated to work with the commerce cache invalidation implementation of the commerce hub (in commerce adapters and in the commerce hub client implementation).

[CMS-19212]

Removal: The ForVendor-Methods from the CommerceConnection

Removed methods `com.coremedia.livecontext.ecommerce.common.CommerceConnection#getServiceForVendor` and `com.coremedia.livecontext.ecommerce.common.CommerceConnection#getQualifiedServiceForVendor`. Use class `com.coremedia.blueprint.base.livecontext.ecommerce.common.CommerceServicesForVendor` instead.

[CMS-19124]

Simplified Registration of Custom Commerce Bean Types

The class `com.coremedia.livecontext.ecommerce.common.CommerceBeanTypeRegistry` was removed and `com.coremedia.livecontext.ecommerce.common.CommerceBeanType` was converted from an interface to an implementation class. Instead of registering enum classes defining custom commerce bean types you may now define custom commerce bean types on demand by calling `com.coremedia.livecontext.ecommerce.common.CommerceBeanType#of(String)`. This solves the issue of class loading order which occasionally lead to missing custom commerce bean types at runtime.

Follow [Section “Simplified Registration of Custom Commerce Bean Types” \[238\]](#) for upgrade information.

[CMS-17159]

Removal: CommerceConnection#getClone Method

Removed `com.coremedia.livecontext.ecommerce.common.CommerceConnection#getClone`. Commerce Connection instances must not be modified after creation.

[CMS-15229]

CoreMedia Headless Server

GraphQL Java Libraries Updated to Version 17.3

These libraries have been updated:

Follow [Section “GraphQL Java Libraries Updated to Version 17.3” \[239\]](#) for upgrade information.

[CMS-20151]

Swagger Updated to Latest Version

Swagger and the related libraries were updated for maintainability reasons. The updates made some refactorings necessary. As a consequence, the previously public api package `com.coremedia.caas.web.rest` has been revoked, as it doesn't contain any classes anymore. The previous public 3rd party package `springfox` has been revoked also, as the CoreMedia classes do not expose the Springfox annotations anymore.

Follow [Section “Swagger Updated to Latest Version” \[240\]](#) for upgrade information.

[CMS-19836]

Refactoring of Adapters

Due to an internal refactoring, some of the adapters used in the content-schema had to be refactored as well. Where possible we tried to be non breaking. Nonetheless some changes lead to breaking changes in the content-schema:

Follow [Section “Refactoring of Adapters” \[240\]](#) for upgrade information.

[CMS-19796]

Filter Predicates May Now Access the DataFetchingEnvironment

Filter predicates were formerly a simple Predicates. They now use a dedicated interface which marks them as ExtensionPoints and they act as a BiPredicate.

[CMS-19769]

Headless Server Deprecations for CoreMedia CMCC 11

These previously announced deprecations were carried out with the major release of CoreMedia CMCC 11:

Follow [Section "Headless Server Deprecations for CoreMedia CMCC 11" \[241\]](#) for upgrade information.

[CMS-19685]

New Feature: Plugin Extension Point FilterPredicate

With the new extension point `FilterPredicate` it is possible to add filter predicate via the means of a plugin. This change is considered breaking, as customers/implementers have to use the new interface `FilterPredicate` instead of the more generic `Predicate<Object>`. `FilterPredicate` is just a marker interface, so the necessary changes are minimal.

[CMS-19641]

`AbstractDynamicListAdapter#getItems` and `SearchServiceAdapter#search` now return `DataFetcherResults`

`AbstractDynamicListAdapter#getItems` and `SearchServiceAdapter#search` now return a `DataFetcherResult`, which wraps data and error information. This enables a uniform way to transfer information about errors from the backend to the client via the `errors` field in the GraphQL response.

[CMS-19428]

Removal: Deprecated Headless API for Commerce Entities

The old commerce schema extension of the CoreMedia Headless Server was deprecated in 2101 and is now removed.

Follow [Section "Removal: Deprecated Headless API for Commerce Entities" \[242\]](#) for upgrade information.

[CMS-19322]

Removal of `ViewBySiteDataFetchingEnvironment`, Public API Changes of `RemoteLinkWiringFactory` and `RemoteLinkDataFetcher`

The class `ViewBySiteDataFetchingEnvironment` has been removed. Use builder of `DataFetchingEnvironmentImpl.newDataFetchingEnvironment()` instead. The `RemoteLinkWiringFactory` and the `RemoteLinkDataFetcher` take a `remoteLinkDataLoader` as constructor argument. The method `CaasWebConfig#getForwardHeaderMap` has been removed. Retrieve the map from the environment context instead.

[CMS-19155]

Addition: New `RichText` Type to Encapsulate the Different Representations of `RichText`

Introduced a new `RichText` type in the schema to encapsulate the different representations of `RichTexts`, like text as tree and linked contents in `RichText`. The old fields are

deprecated and moved into the new type. The field `detailTextLegacy` and `teaserTextLegacy` were introduced to provide easy access to the former behavior of the `detailText` and `teaserText` fields.

[CMS-19107]

Changed Configuration Properties

The following configuration properties have changed and are now defined in the class `CaasSearchConfigurationProperties`:

Follow [Section “Changed Configuration Properties” \[243\]](#) for upgrade information.

[CMS-19076]

Ordered Content Schema

The content-schema file was ordered alphabetically by types, in order to raise the readability of the schema. Since the changed order does not change the code, this change does not break directly. However, customers who changed the content schema directly in that file (Blueprint), will most likely encounter a merge conflict.

[CMS-19071]

Site Query by Root Segment

Starting with 2104.1, the site query now additionally supports 'rootSegment' as a query parameter as an alternative to 'siteId'. The new feature is considered breaking, if the `pageByPathAdapter` was used! `PageByPathAdapter` was renamed to `ByPathAdapter` and the implementation to fetch a site object was moved there. The old implementation in `ContentRoot` was deprecated.

[CMS-18930]

Loading of Persisted Queries from Plugins

Headless server supports loading resource files containing persisted query definitions from the content of plugins.

Follow [Section “Loading of Persisted Queries from Plugins” \[244\]](#) for upgrade information.

[CMS-18315]

Content Types in Search are Now Validated

While searching, non existent content types were simply ignored, leading to the situation, that the search wasn't restricted to any content type at all. As this behavior led to bigger result set than expected and may hide bugs on the client side, a content type validation was introduced. Whenever an invalid, non existent content type is given, headless will respond with one GraphQL error message per invalid content type.

Follow [Section “Content Types in Search are Now Validated” \[245\]](#) for upgrade information.

[CMS-17298]

CMPProductTeaser change in headless

CMPProductTeaser#teaserTarget and CMPProductTeaser#teaserTargets always return null since only the CMPProductTeaser#productRef should only be used. The teaser target can not be set in Studio anyways.

Follow [Section "CMPProductTeaser change in headless" \[245\]](#) for upgrade information.

[CMS-20193]

Headless Commerce: Remove Autocompletion for Breadcrumb Parameter

The breadcrumb parameter is not completed automatically anymore, if the category "root" is missing. Make sure that the breadcrumb parameter contains all categories from leaf to root category, including the root category itself.

Follow [Section "Headless Commerce: Remove Autocompletion for Breadcrumb Parameter" \[245\]](#) for upgrade information.

[CMS-20085]

JsonPreviewClient was refactored to Spring AutoConfiguration

The JsonPreviewClient was refactored to use Spring AutoConfiguration. As a consequence class `com.coremedia.blueprint.headlessserver.JsonPreviewClientConfig` was removed from blueprint. The removal is breaking, if `JsonPreviewClientConfig` was adapted by developers. If not, the change is non-breaking, as `JsonPreviewClientConfig` only contained a component scan which was replaced by the new AutoConfiguration.

[CMS-19941]

Removal: ViewMapperConfig Class

The Spring Configuration class `com.coremedia.caas.web.ViewMapperConfig` was removed from public API. The change didn't affect any configuration properties. The config was meant to expose the existing view mappings during runtime, as the mapping may change during runtime by editing, removing or deleting the homepage of a site. As a configuration class is not meant to expose dynamic runtime data, the class was refactored. The mappings can now be acquired via an actuator at `/actuator/siteRestrictedEndpoints`.

[CMS-19885]

Aligned Behavior of StructAdapter to Same as in SettingsAdapter

When querying for a settings path via StructAdapter, the behavior varied from the one implemented in SettingsAdapter. The behavior of the StructAdapter was now aligned to the behavior of the SettingsAdapter. The documentation was updated accordingly.

[CMS-19452]

Default Java heap for Headless Server reset to 1 GB in Dockerfile

This change only affects customers coming from CoreMedia versions 2101.1, 2101.2 and 2101.3 who use the Docker deployment. The default Java heap for the Headless Server has been reset from 2048m to 1024m in the Dockerfile of the Headless Server. The Java heap can, but does not have to be increased for performance optimization.

[CMS-19210]

Headless Server Schema Changes

Replace the usage of `Locale::toString` with `Locale::toLanguageTag` for the locale fields. This changes the behavior of the locale-field of the Site and CMLocalized types by changing the format of the locale field. Additionally removing the deprecated "languageId" field from the Headless Server schema.

[CMS-19096]

Support ContentBlobLinkComposer in RichText conversion

Added the `ContentBlobLinkAdapterFactory` to the `EvaluationContext` to be able to use the `ContentBlobLinkComposer` in `RichTextConversion`

[CMS-19055]

Miscellaneous

The site-manager container image is now also build with google jib

Because of this change, the convenience feature, that the zip will be configured at startup has been removed.

[CMS-20332]

Support for IBM DB2 has been removed

Support for IBM DB2 has been removed

[CMS-20120]

Removal: Deprecated Private API BlobHelper Class

The private API class `com.coremedia.cap.transform.undoc.BlobHelper` has been removed.

Follow [Section "Removal: Deprecated Private API BlobHelper Class" \[248\]](#) for upgrade information.

[CMS-20049]

Removal: Deprecated Editorial Comments cache properties `editorial.comments.data.cache.*` Properties

The deprecated configuration properties `editorial.comments.data.cache.*` have been removed.

Follow [Section “Removal: Deprecated Editorial Comments cache properties `editorial.comments.data.cache.*` Properties” \[248\]](#) for upgrade information.

[CMS-20047]

Removal: Deprecated `SettingsPageResourceBundleFactory` Class

The deprecated class `com.coremedia.blueprint.cae.web.i18n.SettingsPageResourceBundleFactory` has been deleted. If you still like it, keep it in your project code. As alternative, it is recommended moving resource bundles to `resourceBundles`, and using `LinklistPageResourceBundleFactory` instead.

[CMS-19999]

Removal: Deprecated `ThemeImporterImpl#legacyExternalLinkToDocumentName` Method

The deprecated method `com.coremedia.blueprint.themeimporter.ThemeImporterImpl#legacyExternalLinkToDocumentName` has been deleted. If you still like it, keep it in your project code. Otherwise, it is recommended using `stringToDocumentName` instead.

[CMS-19995]

Removal: Deprecated `UserProvider2` Interface

The interface `UserProvider2` has been deleted. If you are still using it in your project code, simply switch to `UserProvider`.

[CMS-19988]

Removal: Deprecated Legacy RuleProviders Classes

`com.coremedia.blueprint.base.links.impl.PropertiesRuleProvider` and `com.coremedia.blueprint.base.links.impl.SitesRuleProvider` have been deleted from the Java API, incl. the `bpbase-links-legacy.xml` configuration file. You probably do not use the classes directly in your project code. Instead, check if you still import the legacy configuration file in the `blueprint-links.xml` file in the `cae-base-lib` module, and eventually migrate to the `AbsoluteUrlPrefixRuleProvider`.

[CMS-19987]

Removal: Deprecated `LdapMember#getOrganizationalUnit` Method

We have finally deleted `LdapMember#getOrganizationalUnit`, which has never been used. If you have implemented the `LdapMember` interface, you can delete the method. You must at least delete the `@Override` annotation.

Follow [Section “Removal: Deprecated `LdapMember#getOrganizationalUnit` Method” \[250\]](#) for upgrade information.

[CMS-19986]

Removal: `management.health.contentRepository.enabled` Property

The deprecated application property `management.health.contentRepository.enabled` has been removed. Please use its replacement `management.health.uapiConnection.enabled`.

[CMS-19962]

Migrated Blueprint Content Hub Adapters to Plugins

The extensions for the Content Hub Adapters

Follow [Section “Migrated Blueprint Content Hub Adapters to Plugins” \[250\]](#) for upgrade information.

[CMS-19911]

Dropped supported deprecated databases, servers, and browsers

With CMCC v11 the following deprecated databases and servers are not supported anymore: IBM DB2, MySQL 5.7, Postgres 9.6, MongoDB 3.6, and Microsoft Windows Server 2016 Standard.

Follow [Section “Dropped supported deprecated databases, servers, and browsers” \[251\]](#) for upgrade information.

[CMS-19867]

New API For Custom Workflow Extension Point

Accompanying the new Workflow app, there is a new API to add custom workflows. The former API was targeted specifically at the Control Room in the main Studio. The new API can be used to customize both the main Studio and the Workflow App.

Follow [Section “New API For Custom Workflow Extension Point” \[251\]](#) for upgrade information.

[CMS-19858]

Removal: Corporate Extension Modules for Content Server

The [content] server modules for the *corporate* extension were empty and have been removed:

Follow [Section "Removal: Corporate Extension Modules for Content Server" \[252\]](#) for upgrade information.

[CMS-19851]

Removal: Unused class SiteToStoreContextCacheKeyWithTimeout

The deprecated class `com.coremedia.blueprint.base.livecontext.ecommerce.common.SiteToStoreContextCacheKeyWithTimeout` was part of the legacy Blueprint commerce integration and has been removed in favour of the Commerce Hub integration.

[CMS-19850]

New API for Content Type Localization

To support content type localization in both the Main Studio App and the Workflow App (and further apps in the future), we fundamentally changed the API to localize doctypes. One particular change is the direct support for SVG icons.

Follow [Section "New API for Content Type Localization" \[252\]](#) for upgrade information.

[CMS-19816]

Removal: Old live context settings struct

The old 'livecontext.store.config' StructProperty was removed for all commerce Hub enabled sites, tests and demo data. Please use only the new struct below the 'commerce' struct. For more details see JavaDoc for CommerceSettings.java. In addition LiveContextCloneSitePostProcessor.java and it's usages were removed completely, the CommerceCloneSitePostProcessor.java is used instead. Please note: the cms catalog is not affected by this refactoring as it's not a commerce hub enabled site.

[CMS-19396]

Removal: Deprecated API of TransformImageService

The following methods have been deleted from the Java API:

Follow [Section "Removal: Deprecated API of TransformImageService" \[253\]](#) for upgrade information.

[CMS-19254]

Logging Standardization

The logging setup has changed to implement a more standardized setup. This includes:

Follow [Section "Logging Standardization" \[253\]](#) for upgrade information.

[CMS-19224]

Removal: Optimizely Extension

The extension `Optimizely` has been removed from applications `CAE` and `Studio-Client`. The settings starting with "`optimizely.`" have no effect anymore and can be removed from custom content. Additionally the helper class `ExtensionAspectUtil.java` has been removed.

[CMS-19060]

New Blueprint Parent

A new parent POM `cms-blueprint-parent` has been introduced and is used as (transitive) parent for every Blueprint module. It contains basic plugin management, especially for the install- and deploy-plugin for which it is important that all modules use the same versions. If you need a custom parent in your Blueprint, you can simply search and replace our parent element. You could use the `cms-blueprint-parent` as parent of your custom parent, or integrate the `pluginManagement` as necessary.

[CMS-18934]

Replaced deprecated `org.junit.Assert.assertThat`

All remaining occurrences of the deprecated `org.junit.Assert.assertThat` have been replaced by the equivalent `org.hamcrest.MatcherAssert.assertThat`.

[CMS-18795]

Replaced Guava Function and co. with JDK counterparts

Guava `Optional`, `Function`, `Predicate`, `Supplier` and most Guava utilities from `com.google.common.collect` have been replaced with their JDK counterparts. Most of these Guava features are available in the JDK by now, so it is unnecessary to bother with a third-party library. Although it is not required, we would recommend that you also refactor these Guava usages in your code for consistency.

[CMS-18659]

Removal: Deprecated Consumer, Function, Predicate Classes

The classes `com.coremedia.common.util.Consumer`, `com.coremedia.common.util.Function` and `com.coremedia.common.util.Predicate` have been deleted from the Java API. All usages have been replaced with the according JDK classes of package `java.util.function`.

Follow [Section "Removal: Deprecated Consumer, Function, Predicate Classes" \[256\]](#) for upgrade information.

[CMS-18658]

Implementations of Comparable: Added Type Arguments

For several parts of the public API the implementation of the `Comparable` interface now uses correct generic type arguments. As part of this change, following interfaces/classes have been updated by adding type parameter.

Follow [Section "Implementations of Comparable: Added Type Arguments" \[257\]](#) for upgrade information.

[CMS-17548]

Consolidate Blueprint groupIds and versions

The Blueprint now uses the groupId `com.coremedia.blueprint` for every module and Blueprint modules are now always referenced by using `$(project.version)`. The following groupIds are no longer used:

Follow [Section "Consolidate Blueprint groupIds and versions" \[257\]](#) for upgrade information.

[CMS-17559]

4.5.2 Changes and Improvements

CoreMedia Core

Deprecation: Obsolete classes in Workflow Java API

The following classes for customizing workflows have been deprecated in the Java API in package `com.coremedia.workflow.common.util`:

- `AbstractExternalManager`
- `LongActionRegistry`
- `RegisteringLongAction`
- `SpringCollectiveLongAction`
- `SpringDelegateAction`
- `SpringDelegateLongAction`

The methods `hasManager` and `getManager` of class `com.coremedia.workflow.WfServer` have also been deprecated.

For details, see the deprecation notices in the API documentation. To summarize, it's recommended to instead use Unified API interfaces to implement custom managers,

and to use `SpringAware*Action` classes for actions that interact with the Spring context.

[CMS-20233]

Added parametric HTTP endpoint to re-indexing CAE Feeder

The CAE Feeder application now has a specific parametric endpoint to trigger a re-indexing without using JMX. The endpoint is available at `:40781/actuator/reindex` and is enabled by default. To perform a partial re-indexing send a HTTP POST request i.e.

```
curl --request POST \
  --url http://localhost:40781/actuator/reindex \
  --header 'Content-Type: application/json' \
  --data '{
    "ids": "1234,5678",
    "contentTypes": "CMArticle,CMPicture",
    "query": "BELOW PATH '\'/Sites/Calista\'\'\'
  }'
```

See the Search Manual, section "Partial Reindexing of CAE Feeder Indices" for more information.

[CMS-19831]

Update of MongoDB Java Driver to version 4.3.0

The thirdparty dependencies `org.mongodb:mongodb-driver-legacy`, `org.mongodb:mongodb-driver-core` and `org.mongodb:bson` have been updated from version 4.2.2 to 4.3.0.

[CMS-19758]

Deprecation: `ListDiff#elementEqual`

In the Java API, method `elementEqual` of class `com.coremedia.translate.workflow.ListDiff` has been deprecated. Instead of overriding that method in a subclass, you can instead pass a predicate for equality to a constructor of that class.

[CMS-19502]

Removal: Deprecation for `SiteModel`'s `siteIndicatorDepth` in Multi-Site

We removed the deprecation notice for `siteIndicatorDepth` as part of the `SiteModel`.

As part of this change `cm validate-multisite` got extended by a new issue code:

- `MS-VALIDATION-4009 - DISCOURAGED_PROPERTY_VALUE`

The new issue-code is reported at level `INFO` and is triggered, when using a `siteIndicatorDepth` different to `0` (zero).

Motivation

As part of CMS-9521 we deprecated `siteIndicatorDepth` for 2001.1 and changed the default to 0 (zero) rather than 1 (one), as we got aware that many editors struggled with site indicators being hidden in the "depth" of a site – which in the end caused corrupted sites.

Nevertheless, migrating existing content from a depth 1 to depth 0 may raise considerable migration effort, as you have to move all site indicators up one level in one step (while stopping editorial work) and then republish all your sites.

Because of this, we decided to remove the deprecation notice, so that we will support site indicator depths different to 0 also on in the future. The default which ships with CoreMedia Blueprint will stick to the recommended default 0, though.

[CMS-19378]

Third-Party Update: Apache Tika and Transitive Dependencies

Apache Tika has been updated to version 1.26. As part of this change, the following transitive dependencies of Apache Tika have been updated to match versions used by Tika. These updates fix known security vulnerabilities: CVE-2021-28657 (Tika), CVE-2021-27807 (PDFBox).

Updated dependencies:

- com.google.guava:guava: 30.1.1-jre
- org.apache.commons:commons-lang3: 3.12.0
- org.apache.pdfbox:fontbox: 2.0.23
- org.apache.pdfbox:pdfbox: 2.0.23
- org.apache.pdfbox:pdfbox-tools: 2.0.23
- org.apache.pdfbox:preflight: 2.0.23
- org.apache.pdfbox:xmpbox: 2.0.23
- org.apache.tika:tika-core: 1.25
- org.apache.tika:tika-parsers: 1.25
- org.bouncycastle:bcmil-jdk15on: 1.68
- org.bouncycastle:bcpkix-jdk15on: 1.68
- org.bouncycastle:bcprov-jdk15on: 1.68
- org.ow2.asm:asm: 9.1
- org.tukaani:xz: 1.9

If you use these libraries in project code, please check their respective release notes for changes and upgrade information. No changes were necessary in the CoreMedia Blueprint for these updates.

[CMS-19202]

cm dump: Dump Text Blobs to Console

`cm dump` is now able to dump text blob values directly to console. Thus, if your object to dump provides a blob property and it stores some text (or alike) value (such as `application/javascript`), you can trigger to write it to console by using:

```
--blob text
```

The default remains as is, i.e. blobs are just providing some details on MIME type and size. This is the same as calling `cm dump` with:

```
--blob plain
```

[CMS-18880]

Added endpoint to disable/enable the replicator service

The replication-server application now has a specific endpoint to disable and enable the replicator service without using JMX. The endpoint is available at `:8081/actuator/replicator` and is enabled by default. To disable replication send a HTTP POST request i.e.

```
curl -X POST \
  -H "Content-Type: application/json" \
  -d '{"enable": false}' \
  http://localhost:8081/actuator/replicator
```

[CMS-18545]

Changed Content Feeder Configuration Properties

Some configuration properties of the *Content Feeder* have been renamed as part of an initiative to use Spring Boot's type-safe `ConfigurationProperties`. The old property names still work as before but some have been deprecated. It's not required to change your configuration now. If possible, we recommend to migrate to the new property names.

The following properties have been renamed

- `feeder.indexDeleted` -> `feeder.content.index-deleted`
- `feeder.indexGroups` -> `feeder.content.index-groups`
- `feeder.indexNameInTextBody` -> `feeder.content.index-name-in-textbody`
- `feeder.indexReferrers` -> `feeder.content.index-referrers`

- `feeder.partialUpdate.aspects` -> `feeder.content.partial-update-aspects`
- `feeder.management.user` -> `feeder.content.management.user`
- `feeder.management.password` -> `feeder.content.management.password`
- `feeder.backgroundFeed.delay` -> `feeder.content.background-feed-delay`
- `feeder.retryConnectToIndexDelay.seconds` -> `feeder.content.retry-connect-to-index-delay` (new property takes milliseconds if not specified otherwise)
- `feeder.updateGroups.immediately` -> `feeder.content.update-groups-immediately`

See also section 'Content Feeder Renamed Properties' in the reference chapter of the Search Manual for all deprecated and new property names.

[CMS-17957]

Enhanced Encryption Algorithm For System Passwords

The encryption approach used by `cm.encryptpasswordproperty` has been enhanced, so that you may use a more secure `Cipher` transformation to encode passwords than the default `RSA` without padding schema.

To use a better transformation such as `RSA/ECB/OAEPWithSHA-256AndMGF1Padding`, you may now add it as configuration option to your keystore properties file via `CM_CIPHER_TRANSFORMATION`, like:

```
CM_KEYSTORE_PASSWORD=<KEYSTORE_PASSWORD>
CM_KEY_ALIAS=<KEY_ALIAS>
CM_KEY_PASSWORD=<KEY_PASSWORD>
CM_CIPHER_TRANSFORMATION=RSA/ECB/OAEPWithSHA-256AndMGF1Padding
```

For upgrade scenarios, a fallback exists for decoding passwords from old `RSA` transformation, so that you may update your passwords one by one.

To re-encode your passwords, use `cm.encryptpasswordproperty` as described in *Content Server Manual*, section *Password Property Encryption* once you have set `CM_CIPHER_TRANSFORMATION`.

[CMS-17851]

Removal: `com.coremedia.elastic.core.test.Injection` and `core-test` module

`com.coremedia.elastic.core.test.Injection` and `core-test` module is removed. use `@InjectMocks`, `@Mock` and `@Spy` and if this is not possible `org.springframework.test.util.ReflectionTestUtils`

[CMS-16834]

New Validator: PossiblyMissingMasterReferenceValidator for Multi-Site

In certain scenarios it may happen, that you have a master content and derived content which miss a corresponding link from derived to master to signal the relationship. This state, if caused by accident, may cause duplicated contents to be created where you don't expect them.

The new validator `PossiblyMissingMasterReferenceValidator` detects such a state and will by default raise a warning in these states.

Depending on the perspective (if this content is regarded as master content or as derived content) you will get slightly different messages:

- For master you may get: *A content in derived site German (Germany) exists, which may miss a link to this content as its master: Orphaned Name.*
- For derived content you may get: *A content in master site English (United States) exists, which should possibly be linked to as master: Orphaned Name. Consider setting an appropriate master version carefully. Most likely the current master version.*

If for any reason, this analysis reports a false-positive, it will vanish as soon as you finished a translation of a given master content.

Upgrade Notice

No upgrade steps are required unless you consider these warnings irrelevant in your setup. In this case you may remove the bean `possiblyMissingMasterReferenceValidator` which comes with `com.coremedia.cms.middle.blueprint.validators.ValidatorsConfiguration`.

As an alternative you may customize the severity by setting `possiblyMissingMasterReferenceFromMasterValidator.severity` to `INFO` (it defaults to `WARN`).

Extension Points

By default, the match strategy to search for similar contents is done by relative path. If you have different identity tokens (such as product IDs as property values) you may consider replacing the bean. A dedicated section exists in JavaDoc of `com.coremedia.rest.cap.validators.PossiblyMissingMasterReferenceValidator` explaining what possible extension points you may use.

validate-multisite

This validator covers issues detected by `cm` tool `validate-multisite`:

- `MS-VALIDATION-6001` - Content not Used in Derived Site but some Content Exists

- `MS-VALIDATION-6003` – Content without Master but some Content Exists

[CMS-16043]

validate-multisite Improvement

The output data for "MS-VALIDATION-8000 - Link Spanning Sites" has been enhanced with the site IDs of the content's site and the link target's site.

[CMS-11136]

CoreMedia Content Application Engine

Sitemap: Reduce Downtime by Improving the Recreation Strategy.

On recreation the sitemap has been removed before creating the new one. This has been changed. The new one will be created to a temporary folder first and afterwards the old one will be replaced.

[CMS-19719]

Cache-Control Strategy can use Full Request Paths

The `DefaultCacheControlStrategy` can now be configured to use the full request path for pattern matching. When a request contains a semicolon in the path everything after a semicolon is cut in advance of applying the "for-url-pattern". This can be avoided by setting the "`cae.cache-control.urlPath.removeSemicolonContent=false`" configuration property. Additionally the "`cae.cache-control.urlPath.useAlwaysFullPath`" and "`cae.cache-control.urlPath.urlDecode`" configuration properties can be used to adapt the taken request URI path.

[CMS-19195]

Configure CORS Allowed Origins with Studio Host for Preview CAE

The documentation of the `cae.cors.allowed-origins-for-url-pattern` configuration property has been extended with a suggestion to configure it with the Studio host for preview CAE. Furthermore the default Studio host was added to the `cae.cors.allowed-origins-for-url-pattern` configuration property in the `application-dev.properties` of the preview CAE.

Please note that no similar configuration has been added to the default `application.properties` for the preview CAE, because we cannot anticipate allowed origins for production environments.

[CMS-18954]

It is now possible to run the CAE without setting a context path

The CAE context path can now be set to the empty path as follows: ``server.servlet.context-path=/'`. The default was not changed and still uses `"/blueprint"`.

[CMS-17119]

Fixed handling of cyclic navigation structures in CAE

Fixed a bug where the CAE failed to render any page of a site whose root navigation was part of a cyclic navigation structure. For a cycle that contains the root navigation a better to understand error message is now rendered. Cyclic links on other hierarchy levels do not lead to broken previews anymore. A Studio validator and corresponding log messages are available in any case.

[CMS-16628]

Deprecated Header and Footer in CMChannel

All contents of a page are maintained in the pagegrid meanwhile, therefore the `header` and `footer` properties of the `CMChannel` contentbean are unused and have been deprecated.

[CMS-11194]

CoreMedia Studio Client

Fixed Quick Create Dialog Positioning

Disabled behavior where the quick create dialog always positioned next to the menu item it was invoked from. This lead to problem where the dialog may have been rendered outside the screen for menu items, that were rendered close to the screen border.

[CMS-20293]

User Manager Full Text Search

The user manager user and group search does not search for prefixes only anymore. Instead, the term is matched against the whole name. The prefix search can be re-enabled via `_userManagerBase.PREFIX_SEARCH=true_`.

[CMS-20162]

Site Selector - Sorting by Locale

The new site selector supports sorting by locale.

[CMS-19887]

Update to CKEditor 4.16.1

With the update to the `jangularoo-libs` release 4.1.0, the CKEditor was updated to version 4.16.1. Among others, this release of the CKEditor includes some security-related changes. For more details see:

- <https://ckeditor.com/cke4/release/CKEditor-4.16.1>
- <https://ckeditor.com/cke4/release/CKEditor-4.16.0>

- <https://ckeditor.com/cke4/release/CKEditor-4.15.1>
- <https://ckeditor.com/cke4/release/CKEditor-4.15.0>

[CMS-19657]

Added Support for Empty Quick Create Name

The default name of the content created by the QuickCreateDialog is now read from the *QuickCreateSettings.properties* . To support empty names, the key `_default_newDocumentName_text_` can be overwritten with an empty value. This way, the editor is forced to set a custom name.

[CMS-19596]

Added Support for Multiple External Previews

IPreviewUrlTransformer functions can be applied to the external preview URL now too. Only the protocol, host and port of the transformed URL will be applied to the external preview URL, since the context path will remain the same for all preview CAEs and parameters are not relevant here.

This feature is disabled by default so that only the default preview CAE will be used for the external preview. To enable the *IPreviewUrlTransformer* execution for the external preview URL, change the following flag to *true* :

```
ExternalPreviewImpl.ENABLE_PREVIEW_URL_TRANSFORMATION = true;
```

[CMS-19347]

Restored "Open In Tab" Checkbox for Bulk Uploads

The "Open In Tab" checkbox had been removed when the Jobs framework was introduced. The checkbox has been restored now so that documents are opened after a successful upload. The state of the flag is stored in the local storage of the browser.

[CMS-19149]

Studio Client Now Has Referrer-Policy "same-site"

We added the Referrer-Policy "same-site" to the Studio Client. If needed, this configuration can be changed in the studio-base-app module's index.html.

[CMS-19049]

Start Publication Workflow Button From ActionsToolbar No Longer Includes "My Edited Contents" By Default

Previously, the "Start Publication Workflow" button in the Actions Toolbar included all content items from "My Edited Contents". Now, it only includes the currently displayed content item. If desired, you can go back to the previous behavior via a "Content > Actions Toolbar" user preferences setting.

[CMS-18939]

CoreMedia Studio Server

Multi-Selection Support for Taxonomy Link Lists

The *TaxonomySearchField* supports the flag *multiSelect* now. It can be set to *true* in *TaxonomyLinkListPropertyFields*. As a result, the suggestion picker remains open until ESC is pressed and allows to select multiple tags at once.

[CMS-19942]

iFrame Support for Content Hub Preview

Preview data generated by Content Hub Adapters can be marked as 'html' now. This way, it is possible to embed iframes into the Content Hub Preview, e.g. to embed YouTube videos.

[CMS-19492]

Disabling Notifications

The user can disable notifications in the user settings.

[CMS-19486]

Multiple Filter Support for Product List Documents

Product lists support multiple filters now. Existing product list documents are still compatible with the old format and API. If the commerce API supports multi filter facets, a button is shown within the editor to migrate the existing data to the new multi-facet format. After migration, the new multi facet editor will be shown where the editor can select multiple facets, depending on the category that has been linked for the product list document. It is recommended to convert existing product list contents to the new format. Also note that product lists must be checked in the Studio after being translated because not all configured search facets may be translatable.

Additionally, a new product list validator has been introduced. This validator checks if the persisted queries within the documents are still compatible to the search API of the commerce system. The Studio library can be used to search for these issues.

[CMS-19039]

Configure Studio Forms

An editor can now configure the content forms so that they show only parts the editor is interested in. See the chapter "Hiding Studio Form Components" in the Studio User Manual. See the chapter "Hiding Components on Content Forms" in the Studio Developer Manual to prepare your Studio code for the feature.

[CMS-18985]

Fixed Error thrown for Empty UserManager Properties

Fixed a `NullPointerException` that was thrown when the settings for the user manager were missing in the `application.properties`.

[CMS-18855]

Filtering Issues Along with Categories in Studio Library

In the search view of the Studio library, you can now filter content issues not only by severity but also by a category, like localization issues. Note, that the issues filter is only visible, if the *Content Feeder* indexes issues in Solr. Section "Searching for Content | Content Issues" in the Search Manual describes how to enable issue indexing.

[CMS-18055]

Added Thumbnail Support for Content Hub

The content hub thumbnail view now supports an image preview the asset, if the asset supports one. The optional method `getThumbnailBlob()` has been added to the Content Hub API for Content Hub Items for this.

[CMS-16100]

Added metrics to reflect the number of CMS user sessions

Added metrics starting with `cap.session` representing the number of sessions currently known to the application. The following gauges are available:

- `open` the number of distinct users being logged in
- `idle` the number of distinct users who are currently logged in but for whom the application is not currently serving a request
- `busy` the number of distinct users who are currently logged-in and for whom the application is currently serving a request
- `dirty` the number of distinct users whose sessions are about to be closed

[CMS-15893]

CoreMedia Workspace

Update: JUnit 5, Mockito, PowerMock

We updated the following test libraries:

- JUnit BOM to 5.8.1 including:
 - JUnit 5 [Jupiter] to 5.8.1
 - JUnit Platform to 1.8.1
- Mockito to 3.12.4

- PowerMock to 2.0.9

[CMS-20026]

Prevented creation of empty JAR files

The *maven-jar-plugin* is now configured with `skipIfEmpty=true` in all parent POMs. This prevents build failures or the need to add dummy resources for modules that don't contain any runtime sources or resources - e.g. those that only contain tests. **Please note** that installation and deployment must be skipped individually in its *pom.xml* when such a module is added:

```
<properties>
  <maven.deploy.skip>true</maven.deploy.skip>
  <maven.install.skip>true</maven.install.skip>
</properties>
```

[CMS-19388]

CoreMedia Frontend Workspace

New Supported File Types for Theme Importer

The ThemeImporter in the blueprint supports additionally file types. `webp` and `avif` as images and `wasm` and `json` as static resources. If the files are located inside the preconfigured folders like `img/`, `images/` or `vendor/`, they will be copied into the theme zip file automatically. Otherwise add them to your webpack configuration according to the documentation in the Frontend Manual.

[CMS-19886]

Added Lazy Loading to Images

All images in the frontend blueprint, rendered by Freemarker, got an extra attribute `loading="lazy"` by default. All modern browsers support this lazy loading mechanism, which will increase the render time of websites in the browser.

[CMS-19736]

Allow Standalone JavaScripts Files in Themes

Added a new feature that allows building standalone JavaScript files with Webpack in our themes. More information can be found in the Frontend Developer Manual.

[CMS-18953]

CoreMedia Deployment

Enabled Health Checks for Commerce Adapters

After the replacement of the `dockerfile-maven-plugin` with the `google-jib-plugin` to build the Docker images for the commerce adapters, the images don't contain any health checks directive any more. The health check are now configured in the Docker Compose files.

[CMS-20127]

The `wait-for-*` scripts have been replaced with an environment variable controlled mechanism

Instead of changing the entrypoint chain to call a wait script, the new `coremedia/java-application-base:2.4.0-*` images include a tool that can be configured using environment variables. The tool `docker-compose-wait` is configured by `WAIT_HOSTS` and `WAIT_PATHS` to be set to either a comma separated list of `host:port` pairs or file paths.

The old `wait-for-*` scripts will still be available but usages should be migrated.

[CMS-20125]

Graceful Shutdown for all Spring Boot based Applications

Graceful shutdown as described in the Spring Boot documentation [here](#) has been activated.

[CMS-14653]

CoreMedia Commerce Hub

Removal: Deprecated Properties

Properties that were deprecated have now been removed

- `commerce.hub.cache.capacities.*` and `livecontext.ecommerce.cache.*` use `cache.capacities.ecommerce.*` instead
- `commerce.hub.cache.timeout-seconds.*` and `livecontext.ecommerce.cache.*` use `cache.timeout-seconds.ecommerce.*` instead.

[CMS-20183]

Commerce Hub Invalidation Events Added

CMS applications with commerce client integration can now receive cache invalidation events. With this, commerce items are invalidated in the cache of the client applications if the commerce adapter sends an invalidation event, which may be triggered automatically due to change detection (if applicable) or manually using the commerce adapter's `invalidate` actuator endpoint. This feature only works when the client applications are connected to supporting (new) commerce adapter versions. Please see the release notes of the commerce adapters for further information.

[CMS-19482]

Added Configurable Custom Attributes to Commerce Hub Commerce Beans

Commerce Hub Commerce Beans now expose custom attributes if they are added to the entities on the commerce adapter side. This is currently only supported by the Salesforce Commerce Cloud adapter. JSON attributes can be obtained from the commerce bean via `ClientCommerceBean#getCustomAttributes()` as nested map, or via `ClientCommerceBean#getCustomAttributes(Class<T>)` and `ClientCommerceBean#getCustomAttribute(String, Class<T>)` as an object. Deserialization is performed internally using an `com.fasterxml.jackson.databind.ObjectMapper` if the commerce adapter configures `metadata.custom-attributes-format=JSON`.

[CMS-19420]

Catalog Alias Mapping Accepts Catalog ID

In a multi-catalog scenario the catalogs to be displayed are configured in LiveContext settings. All references to catalog items (products, categories) are using an alias name of the catalog. The alias mapping to concrete catalog names are also configured via LiveContext settings. Alternatively to the catalog name also the external catalog id can be used. In some environments this is a better approach because the name is localized and can change more frequently.

[CMS-19058]

CoreMedia Headless Server

New ExtensionPoint PluginRestController

Starting with CM11, plugins may implement a Spring rest controller by implementing the new extension point `PluginRestController`. Controllers will be accessible via the url prefix pattern `'/plugins/plugin-id/request-mapping'`

[CMS-20361]

Add plugin extension point for custom scalar types

Add a new extension point for scalar types. This enables headless server plugins to introduce custom scalar types.

[CMS-20274]

New Config Option: `caas.stax.maxAliasesForCollections`

The maximum number of allowed aliases in a YAML file to define a rich text view was statically bound to 50. When the actual number of aliases exceeded 50, an error while startup of headless was thrown: `Number of aliases for non-scalar nodes exceeds the specified max=50`

The number of allowed aliases can now be configured using the new config option, e.g. `caas.stax.maxAliasesForCollections=100`

Along with the change, the `sakeyaml` library was updated to the latest version `org.yaml:sakeyaml:1.29` [was 1.26]

[CMS-19959]

New Extension Point for Plugins: `WiringFactory`

The `CaasWiringFactory` is a new extension point for custom `WiringFactory` implementations in plugins. See documentation for details.

[CMS-19926]

Add `fullyQualifiedUrl` field to `CMDownload` type

Add the `fullyQualifiedUrl`-field to the `CMDownload` type in the content schema.

[CMS-19859]

New Fields for Localized Content

The GraphQL interface `CMLocalized` in the content schema now contains these two additional fields to ease up the handling of localized variants of a content object:

- `master`: The direct master content object from which the current object was derived from.
- `localizationRoot`: The topmost content object of the localization hierarchy.

[CMS-19832]

New Plugin Extension Point: `PluginSchemaAdapterFactory`

The new ExtensionPoint `PluginSchemaAdapterFactory` enables developers to provide their own schema adapters to be used in a GraphQL schema extension via a plugin. Please see the headless manual for details.

[CMS-19792]

Adds New Endpoint to Support Filename and Extension in Media URL

Extend the Headless Server schema to provide a new URL for Blobs including the filename and extension and provide a new endpoint for those URLs.

[CMS-19572]

Provide Generic Field for Retrieving in Linked Contents in Structs or Settings

Add the option to the SettingsAdapter and the StructAdapter to retrieve multiple values at ones, without wrapping them in their Structure. This supports using the GraphQL schema to build up the structure of the Setting or Struct. Furthermore it enables a generic field to retrieve contents which are linked in Structs or Settings

[CMS-19480]

Additional Headless Augmentation Calls without Commerce Connection

Augmentation data for products and categories can now be retrieved from the Headless Server without an underlying commerce connection. The former augmentation queries still exist, but the new ones are preferable since there is less risk of performance issues due to a slow commerce connection. In order to get inherited augmentation data along the commerce hierarchy, the new Augmentation API requires the caller to pass in the hierarchy/breadcrumb.

The new Augmentation API can be found below the ContentRoot:

- ContentRoot.categoryAugmentationBySite
- ContentRoot.categoryAugmentationBySegment
- ContentRoot.productAugmentationBySite
- ContentRoot.productAugmentationBySegment

If you want to use the new API, make sure your Livecontext Settings contain the following information: commerce.storeConfig.id commerce.catalogConfig.id commerce.vendor

In case of multi catalog projects a fallback commerce connection may still be necessary since catalog alias mapping needs to be resolved by the connected commerce system.

[CMS-19468]

Query for Taxonomies by path

It is now possible to query for taxonomies by path. More details can be found in the Headless Server Manual.

[CMS-19193]

GraphQL Type PageGrid now Offers Name Property for Page Grid

GraphQL type PageGrid missed to offer the name of the page grid.

[CMS-18889]

MediaController Supports Conversion of Image Formats

The MediaController supports the conversion of images into supported image formats (jpeg, png and gif) by specifying the corresponding file extension with the file name. The conversion is also reflected in the content-schema. It is now possible, to specify an optional image format when querying for uri templates.

[CMS-18878]

New Fields Contain Lists of Content Objects Embedded in Richtexts

Starting with version 2104.1, the content schema offers complementing fields for all rich text fields. For GraphQL type CMTeasable

- detailTextReferencedContent
- teaserTextReferencedContent

For GraphQL type CMMedia

- captionReferencedContent

[CMS-18844]

Custom Filter Queries

Custom Filter Queries are available for Search and Dynamic Query Lists now. Consult the Headless Server Manual for details.

[CMS-18826]

Loading of JSLT and REST Mappings From Plugins

Headless Server supports loading resource files containing JSLT and REST mapping definitions from the content of plugins.

By loading these kind of resources from plugins rather than from within the Headless Server Docker image, the build, deployment and release management of the core application is decoupled from the development of these resource files.

Loading from plugins is optional. The deployment within the Headless Server App remains unchanged.

[CMS-18486]

Pagination for collections

Pagination for collections has been introduced for a set of fields, that return a collection. The new fields with pagination support are:

- Banner: mediaPaged, picturesPaged
- CMChannel: headerPaged, footerPaged
- CMCollection: itemsPaged, teasableItemsPaged

- CMGallery: `mediaItemsPaged`
- CMLinkable: `contentInSettingPaged`, `contentInStructPaged`, `subjectTaxonomyPaged`, `locationTaxonomyPaged`
- CMLocTaxonomy: `locChildrenPaged`
- CMNavigation: `childrenPaged`
- CMTaxonomy: `childrenPaged`
- CMTeasable: `authorsPaged`, `relatedPaged`, `teaserTargetsPaged`, `videosPaged`, `spinnersPaged`

See documentation for details.

Some fields, for which pagination was already implemented, have been deprecated in favor of the new mechanism: `CMQueryList#pagedItems`, `CMQueryList#filteredPagedItems`. Also some methods of `QueryListAdapterFactory` and `QueryListAdapter` have been deprecated for that reason, see javadoc for details.

[CMS-17909]

New Plugin Extension Point: `CopyToContextParameter`

With the new extension point `CopyToContextParameter` it is possible to declare additional parameters (http request parameters and headers) to be copied in the graphql context programmatically.

[CMS-17512]

Miscellaneous

Configurable Log Pattern in Solr Base Image

The Solr base image has been enhanced to make the Solr logging pattern configurable. Use `SOLR_LOG_PATTERN_CONSOLE` and `SOLR_LOG_PATTERN_FILE` to configure the logging.

This fix is contained in `coremedia/solr-base:8.8.2-3` and higher.

[CMS-20097]

Undeprecated `CMLinkable#LOCAL_SETTINGS`

On popular demand, we have undeprecated `com.coremedia.blueprint.common.contentbeans.CMLinkable#LOCAL_SETTINGS`. Nonetheless, we recommend to use the `SettingsService` to access the settings.

[CMS-19985]

Tomcat TLD Scanning has been Disabled for all Applications Except CAE

Since the CAE is the only application using taglibs, the tomcat scanning mechanism has been disabled for all Spring Boot applications except for the CAE.

For the CAE an exclude pattern has been defined, excluding problematic jars. The list includes:

- `xmpcore*.jar`
- `grpc-netty-shaded*.jar`
- `listenablefuture*.jar`
- `jsr305*.jar`
- `checker-compat-qual*.jar`
- `j2objc-annotations*.jar`

Please make sure that you are not already setting:

- the property `server.tomcat.additional-tld-skip-patterns`
- the environment variable `SERVER_TOMCAT_ADDITIONALTLDSKIPPATTERNS`
- the environment variable `SERVER_TOMCAT_ADDITIONAL_TLD_SKIP_PATTERNS`

If you set one of these properties outside of the CAEs `application.properties`, make sure to remove those and merge your additions into the CAEs `application.properties`.

[CMS-19953]

Added Headless Rest Mappings for Augmentation Queries

The headless server now provides an augmentation Rest API. Refer the documentation for more details.

[CMS-19900]

Third-Party Update: Apache Solr and Transitive Dependencies

Apache Solr has been updated to version 8.10.1.

For detailed changes of Apache Solr have a look at https://lucene.apache.org/solr/8_10_1/changes/Changes.html.

In previous releases, the Solr configuration contained a workaround for a bug in Solr 8.8.2, that has been fixed with the new version [SOLR-15192 / SOLR-13034]. The workaround has been removed again in file `apps/solr/modules/search/solr-config/src/main/app/configsets/content/conf/solrconfig.xml`

. For best performance, it's recommended to apply this change to custom copies of that file, by setting the `enableLazyFieldLoading` option back to `true`.

As part of this change, the following transitive dependencies have also been updated in CoreMedia applications to match versions used by SolrJ:

- HttpComponents HttpCore: 4.4.14
- Jetty: 9.4.44.v20210927

If you use these libraries in project code, please check their respective release notes for changes and upgrade information. No changes were necessary in the CoreMedia Blueprint for these updates.

[CMS-19802]

Update of Commerce Adapter Versions

The following commerce adapter versions are set:

- Commerce adapter base to v1.6.9
- Commerce adapter sfcc to v1.4.6
- Commerce adapter hybris to v1.4.3
- Commerce adapter wcs to v1.6.3
- Commerce adapter commercetools to v1.2.3

[CMS-19734]

Third-Party Update: Spring Boot and Related Dependencies

Spring Boot has been updated to version 2.3.11.RELEASE. As part of this change, the following dependencies have been updated to match versions used by Spring Boot. This also fixes a security vulnerability in Spring Security: CVE-2021-22112

- Spring Boot 2.3.11.RELEASE
- Spring Framework 5.2.15.RELEASE
- Spring Security 5.3.9.RELEASE
- Byte Buddy 1.10.22
- Glassfish JAXB 2.3.4
- Hibernate 5.4.30.Final
- Hsqldb 2.5.2
- Jakarta Mail 1.6.7
- Janino 3.1.4
- Jetty 9.4.41.v20210516

- Micrometer 1.5.14
- MySQL 8.0.25
- Netty 4.1.65.Final
- PostgreSQL driver 42.2.20
- Tomcat 9.0.46

If you use these libraries in project code, please check their respective release notes for changes and upgrade information. No changes were necessary in the CoreMedia Blueprint for these updates.

[CMS-19524]

Added the possibility to change the commerce reference of an augmented content

In the case that IDs of categories or products changes in the commerce system, augmentations can now be adapted from the old ID to the new ID directly in CoreMedia Studio. To that end the system tab of augmented contents now provides an editable commerce reference field and a new `change reference` button. By clicking on it, the editor has the possibility to replace the current commerce reference with a new one from the commerce library. Folder path and name of the augmented content is getting updated after the change.

Additionally, the preference "*Show Products and Categories as Content*" also toggles the display of the repository path on the `System` tab of augmented categories and products.

[CMS-19456]

Query for Documents Tagged by Specific Taxonomies

It is now possible to query for documents tagged by specific taxonomies. More details can be found in the Headless Server Manual.

[CMS-19408]

Explode Spring Boot Applications for Commerce Adapters at Startup

The Docker images of the commerce adapters now also make use of the `SPRING_BOOT_EXPLODED_APP` toggle was added to the `java-application-base` image. By setting `SPRING_BOOT_EXPLODED_APP: true` [which now is the default], the `application` entrypoint will extract the `application.jar` to `/coremedia/exploded` and start it from there. If that directory already exists, it will start the application from there.

This process speeds up the starting process by factor 2-3 and is the recommended way by Spring.

[CMS-19368]

Reproducible Maven Builds

With this release the Maven build produces the same artifacts on two consecutive builds, when there are no changes. This is implemented following the Maven guidelines described [here](#).

There are some minor behavioral changes in the editorial UIs:

- The Studio won't show a build date in the about dialog if the version is a `-SNAPSHOT`.
- The Editor won't show a build date in the about dialog

The `coremedia-application-maven-plugin` has been updated with this commit to the version `3.0.0`. This version does not add any new configuration options but removes some unused functionality including:

- support for Webstart
- support for RPM building
- support for application building with embedded Tomcat installation

If you need any of these features, you can switch back to version `2.8.3` and build without the reproducible build feature.

[CMS-19166]

Reworked Commerce search API

The Commerce search API as defined in class `com.coremedia.livecontext.ecommerce.catalog.CatalogService` was changed. The methods accepting search parameters as a map are deprecated. A single method accepting parameters of type `com.coremedia.livecontext.ecommerce.search.SearchQuery` and `com.coremedia.livecontext.ecommerce.common.StoreContext` replaces them all: `CatalogService#search(SearchQuery sq, StoreContext sc)`

The paging parameters have been consolidated. The new `SearchQuery` parameter object does neither support `pageSize` nor `pageNumber` because this information is redundant to `limit` and `offset`.

The new `SearchQuery#offset` is 0-based whereas `CatalogService#SEARCH_PARAM_OFFSET` was 1-based. In order not to confuse the editors the offset in the `ProductList` Studio form is still 1-based. The required offset conversion is implemented in all places, where the settings of product list contents are read. Custom implementations will have to be changed from `searchParams.put(SEARCH_PARAM_OFFSET, Integer.toString(myOffset))` to `searchQueryBuilder.setOffset(myOffset - 1)`.

The method `MarketingSpotService#searchMarketingSpots(String s, Map<String,String> m, StoreContext sc)` has been deprecated

in favor of `MarketingSpotService#searchMarketingSpots(SearchQuery q, StoreContext sc)`.

The methods `DataSearchService#searchCategories(String s, Map<String,String> m, StoreContext sc)`, `DataSearchService#searchProducts(String s, Map<String,String> m, BaseCommerceBeanType b, StoreContext sc)` and `DataSearchService#searchMarketingSpots(String s, StoreContext sc)` have been deprecated in favor of `DataSearchService#search(SearchQuery q, StoreContext sc)`.

Multi-facet support is included only for commerce adapter services based on base adapter 1.5. The new search implementation is backward-compatible in the sense that it works with older commerce adapter services (using base adapter older than 1.5) but then just the first facet is taken into account. Also note that the deprecated `SearchResult#getFacets` is only filled when connected to an old commerce adapter because the returned facet format is incompatible with the new facet format returned in `SearchResult#getResultFacets`.

For the backward compatibility of the commerce adapter used in your project, please consult the release notes of the commerce adapter.

[CMS-19118]

Removed Classic Studio Components

Removed Components in `apps/studio-client/apps/main/core/editor-components/src/sdk/desktop/`:

- `ExtensionsMenu/ExtensionsMenuToolbar` replaced by the new `Apps Menu`
- `FavoritesToolbar/FavoritesToolbarUserItemsArea` is no longer a toolbar, but a menu: `FavoritesButtonMenu`, `FavoritesUserItem`

[CMS-18876]

Added `EntityController` and `StudioRestController` as `ExtensionPoints` for Studio-Server Plugins

See JavaDoc of `EntityController` and `StudioRestController` in package `com.coremedia.rest.controller` and section "Using the Entity-Controller" in Studio Developer Manual and section "Plugins for Java Applications" in Blueprint Developer Manual.

[CMS-18735]

Unify Link Building for Commerce-Led and Content-Led URLs

The generic client receives links for commerce-led and content-led link building as link template. The template usually contains placeholders, which are replaced in the generic client.

The class `com.coremedia.livecontext.ecommerce.link.CommerceLinkTemplateTypes` represents the list of well-known link templates. As all link templates are transferred in a map from the adapter to the client, additional links can be added on the adapter and client side easily without API changes.

Make sure the commerce adapter instance and the commerce-adapter-base library in your client application do have at least version 2.0.0.

There is no further action needed, if you did not customize link building in your project so far.

[CMS-17506]

Customization of `CMQueryListImpl` and `SettingsStructToSearchQueryConverter`

The Blueprint classes `CMQueryListImpl` and `SettingsStructToSearchQueryConverter` can be customized easier now by overriding protected methods, that were private before. The `SettingsStructToSearchQueryConverter` was enhanced by a method `applyCustomFilter` that can be overridden to add custom filters to the query. Additionally a custom `SettingsStructToSearchQueryConverter` can be provided via a new factory bean `SettingsStructToSearchQueryConverterFactory`.

[CMS-16276]

4.5.3 Fixed Issues

CoreMedia Core

Fixed Bug in `LocalAndLinkedSettingsFinder`

Class `com.coremedia.blueprint.base.settings.impl.LocalAndLinkedSettingsFinder` handles cyclic navigation structures and does no longer propagate `com.coremedia.blueprint.base.tree.CycleInTreeRelationException` to clients of `com.coremedia.blueprint.base.settings.SettingsService`.

[CMS-20313]

Update of coremedia-application-maven-plugin to 3.0.1

This update fixes an issue, where the plugin did not set the executable flags correctly for the `cm` executable in the `target` directory.

[CMS-19862]

Third-Party Update: Spring-Security

Due to a security issue [<https://nvd.nist.gov/vuln/detail/CVE-2021-22119>] we updated spring-security from 5.3.9.RELEASE to 5.3.10.RELEASE. We are not aware of any incompatibilities.

[CMS-19702]

Fixed CAE Feeder deadlock for JMX access

Fixed a bug in the *CAE Feeder* that could lead to a deadlock when reading attribute `InvalidationQueueSize` of `MBean ContentDependencyInvalidator` via JMX.

[CMS-19642]

Third-Party Update: Apache PDFBox

Apache PDFBox has been updated to version 2.0.24 to avoid a security vulnerability of the previous version [CVE-2021-31812].

[CMS-19638]

CAE Feeder Performance Improvement

Fixed a bug in the *CAE Feeder* that led to reduced throughput, especially for larger indexes.

[CMS-19636]

Fixed Output for `cm encryptpasswordproperty`

`cm encryptpasswordproperty` failed to output more details especially in case of failures while encrypting or decrypting passwords.

This has been fixed.

Upgrade:

Unless you defined a custom `EncryptionService`, no upgrade steps are required.

For custom implementations of `EncryptionService` it is strongly recommended to override the default method `EncryptionService.usingLogger(logger:Logger):EncryptionService`, so that tools such as `cm encryptpasswordproperty` benefit from log messages provided by your custom service. Ensure to read the implementation specification carefully.

As alternative, you may want to adjust the tools-logging-configuration, so that your custom service is able to log to stdout.

[CMS-19531]

The 'property' column of the 'System' table has now the length 100

When a string property of a document type is observed it is registered in the 'System' table. To handle long document type and property names the 'property' column of the table has now the length of 100 instead of 50.

[CMS-19293]

Verbosity of publishall

Failures of the `publishall` tool are logged to a file, but have not been obvious for the user on stdout. Now, a concluding failure message is written to stdout.

[CMS-19275]

cm validate-multisite: Enhanced Robustness for Huge Multi-Site Setups

`cm validate-multisite` got enhanced for robustness in huge multi-site setups.

Instead of buffering issues in memory before they are stored to stdout and/or file, the output is now done immediately after an issue has been found.

Due to the change, the output to stdout has changed slightly, so that reported issues are mixed with status messages. For a better overview of actual issues found, it is recommended to dump the results into a tab-separated file using the `--file` CLI option.

[CMS-19272]

Blobs in structs get collected after version or content is destroyed

Previously, blobs that were referenced in a struct did not get garbage collected when the version containing the struct that was destroyed. This has been fixed.

[CMS-19160]

Fix import-user script if no user xml files found

The `import-user` script in the management-tools container now ignores an empty users dir in the import directory.

[CMS-19103]

Fixed Broken Order of Transformations

The list of image transformation variants is now returned sorted by name. This may lead to a different order of variants inside the Studio's Image Editor. To customize the order, change the struct list of variants in the settings document *Responsive Image Settings*.

[CMS-18680]

Fixed Solr hostname and port disclosure with malicious REST request

It was possible to create a manipulated bad Studio search request which produces an internal server error with an error message text produced by Solr containing the hostname and port. This error is caught now.

[CMS-18530]

Improve Studio robustness when no WorkflowRepository is available

When no `WorkflowRepository` was available in Studio some errors occurred. Now all accesses to the `WorkflowRepository` have been guarded by a `null` check to prevent errors.

[CMS-18237]

Fixed multi-threading issues in cleanrecyclebin tool

An issue has been fixed regarding the concurrent creation of directories.

[CMS-18163]

Fixed a bug in ObservedPropertyService leading to seemingly missing augmentations

Fixed a bug in the implementation of `com.coremedia.cap.content.observe.ObservedPropertyService` that lead to wrong cache entries of observed property values. If the cache key was evaluated in a request associated with a user session with limited rights, then the cache entry contained only the contents readable by that user. Because the Augmentation Service is using observed property values to find augmented content for a commerce ID, this could have lead to seemingly missing augmentations in the past.

[CMS-16741]

Encrypt Initial Passwords

The properties `cap.server.initialPassword.*` can be encrypted using the `cm encryptpasswordproperty` tool now.

[CMS-11312]

Fixed Content Feeder Handling of Destroyed Derived Content

Fixed a bug in the Content Feeder that could lead to an error while feeding a content item, if a derived content item was destroyed in the meantime. Exceptions about destroyed derived content items are now caught and do not cause problems for feeding.

[CMS-9058]

CoreMedia Content Application Engine

Removed com.coremedia.elastic.social.springsecurity from component scan

The package `com.coremedia.elastic.social.springsecurity` has been removed from Spring component scan to avoid conflicts with auto configuration and unwanted instantiation of `SocialWebSecurityConfigurerAdapter` when a custom `WebSecurityConfigurerAdapter` is in place.

[CMS-19671]

Cache Classes Listing in DataViewFactory MBean fixed

Fixed displayed cache classes in `DataViewFactory#getCacheSizes` MBean

[CMS-19600]

Avoid Cache Deadlock for TransformedBlobKey

Avoided calling equals on the original blob as that results in a deadlock within the cache. `CapBlobRefImpl` is dereferenced while calling `com.coremedia.transform.impl.CachedTransformedBlob#getETag` which results in avoiding cache deadlock

[CMS-19527]

PageGrid Issue on Category and Product Pages Fixed

In rare cases it could happen that wrong content was delivered from the cache. The reason was a problem with the identity of the `PageGridCacheKey`.

[CMS-19416]

Fixed cache size handling of the DataViewFactory

The default `com.coremedia.objectserver.dataviews.DataViewFactory` bean now respects cache sizes that have already been configured on the cache. If a cache size for a dataview class is defined inside the dataviews XML file as well as by a `cache.capacities` spring property, then the Spring property wins.

[CMS-19244]

CapBlobHandlerTest doesn't fail even if mediatransform.xml is populated

There has been some refactoring since 2101.2 and the bug is not reproducible since 2101.3.

[CMS-18988]

Sanitize user-data before logging it

User data provided is now sanitized before logging the entry to the log files, thus avoiding log injection.

[CMS-18000]

Added hint for TransformedBlobCache usage

Added hint to CAE Developer Manual, that the transformed blob cache can only be used by one CAE.

[CMS-17644]

CoreMedia Studio Client

Window Skin removal

- The Window Skin GRID_200_LIGHT is removed.

Use the Skin GRID_200 instead.

[CMS-20345]

Fix commerce library lazy loading

Reimplemented how is checked whether a category inside the library is a linked one. The new way is more performant and does not load all the category children from the parent.

[CMS-20245]

Fixed Error On Closing CMPicture Document

We fixed an error that occurred when closing a CMPicture document [with tab-reuse disabled].

[CMS-20221]

FolderChooserListViewBase#getAvailablePathsExpression is now null robust

`FolderChooserListViewBase#getAvailablePathsExpression` threw a `TypeError` because of an unallowed `null` access when no preferred site was selected. This occurred while importing an item from the content hub.

[CMS-19707]

Fixed Link List Suggestions for Site Specific Taxonomies

Fixed an issue where only global taxonomy nodes were suggested in the link list, even for site specific ones. The issue happened for the link list suggestions and for the taxonomy chooser dialog chooser.

[CMS-19611]

Added "defaultNameValueExpression" Parameter to QuickCreateMenuItem

There parameter was missing here and is now passed via the *QuickCreateAction* to the *QuickCreateDialog* .

[CMS-19595]

Modification of Issue Filter UI selection

When choosing an issue category, with no severity selected before, all the severity checkboxes are now checked automatically.

[CMS-19545]

Added new config preferredSiteOnly for LinkListPropertyField

The config delegates the config to the LinkListPropertyFieldGridPanel. If true its link suggerster only suggest content of the preferred site.

[CMS-19491]

CategoryDocumentForm Expands when there is an Issue

The subpanels for `subjectTaxonomy`, `locationTaxonomy`, `keywords` will be expanded when there are issues fo the properties.

[CMS-19382]

Fixed Caching Issue in ViewTypeUtil

Fixed caching issue in *ViewTypeUtil.as* when working the custom base paths.

[CMS-19267]

Cache Buster is once again only applied for GET requests

When changing the used underlying API for our Studio REST Requests we accidentally added the cache buster (that prevents caching a requests) not only to "GET" requests but to all requests. This is fixed now, so the old behavior applies.

[CMS-19243]

Fixed some timing problems in FolderChooserList

The FolderChooserList was not properly initialized if some contents were not yet loaded. The has been fixed now.

[CMS-19158]

Possible Format-Loss For Richtext in Studio Fixed

For CoreMedia Richtext created outside CoreMedia Studio you may have experienced losing formatting when loading these into CoreMedia Studio.

The root-cause, which required all CoreMedia Richtext documents to come with a xlink namespace declaration, got fixed.

[CMS-19138]

Fixed type error when aborting a workflow

Fixed a type error when a workflow was aborted in Studio.

```
Uncaught TypeError: this$.backToListHandler$2G1D is not a function
```

[CMS-19133]

Studio Client Now Has X-Content-Type-Options "nosniff" Header

The Studio Client now has the X-Content-Type-Options "nosniff" header set for all client requests. The Studio Server already added this header previously.

[CMS-19050]

Fixed Timing Problem when Loading Site Specific Settings

If global settings were loaded before the site specific settings the global settings sometimes did take precedence. This has been fixed now.

[CMS-18549]

Fixed error when trying to republish a withdrawn folder

A published and then withdrawn folder could not be republished again. An error message appeared indicating that the folder is already published. This is fixed now.

[CMS-17407]

Controlroom: Start translation workflow button in toolbar hidden when unusable

A bug has been fixed which shows the button to start a translation even if it is not possible to start a translation. Now the button is hidden in the case a translation can't be started.

[CMS-16881]

Fixed endless loop bug when opening the Studio library

When a new site was created which only contains two contents of type Site Indicator and CMChannel with the latter being the home page of the former, then it could happen that opening the library results in an endless JavaScript execution loop and a type error being displayed in the browser console which starts with the following error text:

```
AS3: [ERROR] Ext.ComponentManager.register(): Duplicate component id  
"librarytree"  
Error: Duplicate component id "librarytree"
```

[CMS-16737]

Fixed console warnings about deprecated Ext.EventManager

The EventManager is outdated causing a lot of warnings when opening e.g. an CMArticle form due to still being used. Replaced all usages in the code with corresponding alternatives.

[CMS-16061]

CoreMedia Studio Server

Fixed Notification Menu Closing Bug

Fixed an issue where the notification menu was closed when clicking on it's scrollbar.

[CMS-20021]

Mimetype of uploaded xpi files is now recognized as application/x-xpinstall

Prior to the fix they have been recognized as application/zip.

[CMS-19951]

Improve 'mayPreview' of ConfigureDocumentTypePlugin

The *mayPreview* check is now executed for the complete hierarchy of a document type. The check was previously only executed for the first function that matches any type of the given content type.

[CMS-19905]

Fixed Caching Bug for View Types

Fixed a bug in the view type selector where wrong variants have shown or none at all for different document types.

[CMS-19902]

Error for Editing History of Deleted User Fixed

We fixed an error that occurred when you view the editing history of a document where some versions were created by a user that was deleted in the meantime.

[CMS-19733]

Fixed Issue with Tab Expand Plugin

Fixed possible *null* error in the *TabExpandPlugin* which is responsible for the "Advanced Tab" setting in the user preferences.

[CMS-19715]

Fixed Error in Job Tooltip

Fixed a *null* error inside the running job panel.

[CMS-19682]

Reject requests against Studio Server until the application is live and ready

Depending on the deployment circumstances, Studio Server may have accepted requests before the application was live and ready. This lead to broken JSON responses and broken studio client sessions. This bug has been fixed by adding a filter that rejects requests during Studio startup and permits requests when Studio is live and ready.

[CMS-19589]

Fixed initialization of Studio ObjectMapper

Fixed a bug that caused the Studio ObjectMapper to get initialized without the JsonSerializer required by the Studio Client under rare conditions.

[CMS-19582]

Third-Party Update: Liquibase

Liquibase has been updated to 4.3.5.

[CMS-19389]

Fixed Tooltip of Notification Menu Button

The tooltip of the Notification menu button wasn't showing the proper amount of unread notifications. This has been fixed now.

[CMS-19323]

Embedding a CMHTML document into the richtext editor no longer leads to errors

There was a bug when embedding a CMHTML document to the richtext editor causing JavaScript errors and breaking Studio. This has been fixed now.

[CMS-19280]

Theme Importer

The Theme Importer did not recognize changes of the `defer` flag for Javascript resources from `true` to `false` when reimporting a theme. This is fixed now.

[CMS-19279]

Fix StackoverflowError in Studio Server when starting sync workflow

Fixed a bug that caused a StackoverflowError in the Studio Server when a synchronization workflow was started for content with illegal cyclic master link structures.

[CMS-19119]

Fixed Referrer Invalidations

In some situations the order of invalidations regarding referrers of a content was not deterministic. This lead to stale cache entries that still contained outdated referrers. The order of referrer invalidations is now fixed in the U-API client. This especially solves

an error that shows "Cycle detected in tree relation" in the log or a Studio validator message regarding a cycle in the navigation, although the cyclic link was already removed from the content repository.

[CMS-19109]

Enhanced Error Handling For Editorial Comments When Connected to Old Server

Due to missing support for User-UUID support in old content management servers, you may have observed occurrences of `EditorialCommentsPersistenceException` with rather vague errors on column `CREATOR_ID`.

The error handling has been enhanced in that way, that you will get a clear error message for this state: *The Content Server does not provide a UUID for user. This may happen if Content Server version is before 2007.1* [message of a corresponding `NullPointerException`].

[CMS-19086]

Studio Locale Cookie with SameSite Lax

Firefox was complaining about the Studio locale cookie having `sameSite=none` and at the same time not being secure. Firefox also threatens to disregard the cookie in the near future which will result in Studio not loading anymore whenever a locale different from the default locale is stored in a user's `EditorPreferences`.

The cookie now has `sameSite=Lax` from 2101.4 and 2104.1 on. For older versions there is a workaround applicable to the `config-init.js` in Blueprint.

Replace the function `loadScripts()` with this:

```
function loadScripts() {
  // create and append script elements
  function loadScript(attributes) {
    var script = document.createElement('script');
    script.async = false;
    script.type = 'text/javascript';
    Object.keys(attributes).forEach(function (a) {
      script[a] = attributes[a];
    });

    document.body.appendChild(script);
  }

  var attributeList = [
    {src: window.coremediaRemoteServiceUri + 'supported-locales.js'},
    {src: window.coremediaRemoteServiceUri + 'accept-language-header.js'},

    {src: 'resources/before-ext-load.js'},
    {id: 'microloader', 'data-app': '906bf4bf-9a7d-42cc-b7a5-6ef30df325e9',
     src: 'bootstrap.js'}
  ];

  patchJooLocaleSupportWhenReady();

  attributeList.forEach(loadScript)
```



```
}

```

and add:

```
function patchJooLocaleSupportWhenReady() {
  joo.___localeSupport = joo.localeSupport;
  Object.defineProperty(joo, "localeSupport", {
    get: function () {
      return joo.___localeSupport;
    },
    set: function (ls) {
      joo.___localeSupport = ls;

      ls.setLocale = function (newLocale/*:String*//*:String*/ {
        ls.locale = ls.findSupportedLocale(newLocale);
        // either create, update or remove (if locale===null) the Cookie:
        setLaxCookie(ls.localeCookieName, ls.locale, ls.localeCookiePath,
          ls.locale ? getLocaleCookieExpiry() : new Date(0),
          ls.localeCookieDomain);
        return ls.getLocale(); // use getter to re-compute fallback logic
        for locale===null and cache the result
      }

      return ls.getLocale();
    }
  })
}

function setLaxCookie(name, value, path, expires, domain) {
  //noinspection FallThroughInSwitchStatementJS
  document.cookie =
    name + "=" + encodeURIComponent(value || "") +
    ((expires === null) ? "" : ("; expires=" + expires.toGMTString()))
+
    ((path === null) ? "" : ("; path=" + path)) +
    ((domain === null) ? "" : ("; domain=" + domain)) + ";
  SameSite=Lax"
}

function getLocaleCookieExpiry()/*:Date*/ {
  var date/*:Date*/ = new Date();
  var DAYS_TILL_LOCALE_COOKIE_EXPIRY = 10*356;
  date.setTime(date.getTime() + (DAYS_TILL_LOCALE_COOKIE_EXPIRY * 24 * 60
    * 60 * 1000));
  return date;
}

```

[CMS-18892]

Fixed VisibilityValidator bug that caused NoSuchPropertyDescriptorException

Fixed a bug in the Studio VisibilityValidator that could lead to an exception of type `NoSuchPropertyDescriptorException` when issues were computed for a page, and if a link was missing in the page grid structure.

[CMS-17791]

BlobUploadXmpDataInterceptor merging existing local settings now

If local settings are already existing when calling `BlobUploadXmpDataInterceptor`, they will now be merged and won't be overwritten anymore.

[CMS-16028]

Fixed Problem With Vanished Characters in Text Fields / Areas

We fixed a bug where characters vanished in text fields and text areas when one was typing fast.

[CMS-15263]

CoreMedia Elastic Social

RemoveLikesTask updates counter "author:number_of_likes"

The RemoveLikesTask updates the counter "author:number_of_likes" now. Before, the counter was not updated by the task.

[CMS-19773]

SimpleLinkFormatter link building properties configurable and protocol default to HTTPS

ElasticSocialLinkBuildingProperties are introduced to make link building properties (protocol, context path and servlet path) configurable. Default protocol is now set to HTTPS.

[CMS-18425]

CoreMedia Deployment

Fix warning about spring.profiles at application startup

The `coremedia/java-application-base` image accepts a `SPRING_PROFILES` environment variable to enable Spring profiles. With newer versions of Spring Boot, `SPRING_PROFILES` is interpreted as the `spring.profiles` systemproperty which cannot be set as it is a node in the property tree and not a leaf.

With the update to `coremedia/java-application-base:2.4.0-*`, the `SPRING_PROFILES` environment variable will be unset before the application is started. The value of the environment variable is however passed to the start of the application to enable all requested profiles.

[CMS-20015]

File logging can be disabled

File logging could not be disabled for container based environments. This behavior was fixed with a conditional log back configuration. To disable file logging, set environment variable "LOGGING_FILE_NAME" to an empty value.

[CMS-18834]

CoreMedia Commerce Hub

Extend live context environment with fragment connector disabled

The `com.coremedia.fragmentConnector.disabled` property now can be configured via properties files such as `project.properties` or `lc-connector.properties`, besides via Standard Environment variable or JVM argument.

[CMS-19939]

Fixed propagation of CatalogAlias to CommerceBeans' StoreContexts

Commerce beans now propagate the catalog alias which is encoded in their commerce ID to derived commerce beans such as subcategories or assigned products. This fix ensures that the catalog alias is also set to the store context held by the commerce beans.

[CMS-19675]

Fixed delegation to built-in commerce services for custom commerce service implementations

Introduced `com.coremedia.blueprint.base.livecontext.client.common.RequiresServiceDelegate` as callback interface if a custom commerce service implementation requires the built-in service instance as delegate.

[CMS-19453]

Fixed a bug regarding `commerce.hub.cache` properties

The deprecated commerce cache properties prefixed `commerce.hub.cache` caused application startup to fail. This bug has been fixed so that these properties can be used again. But still remember to convert them to `cache.<capabilities|timeout-seconds>.ecommerce` properties soon.

[CMS-19324]

CoreMedia Headless Server

Refactor PluginConfiguration

Remove usage of superfluous "name" attribute in Bean Annotation and introduce constant for bean name for custom SchemaAdapterBeanResolver.

[CMS-20408]

Fix rendering of richtext in settings

Properly invoke the RichTextAdapter in the Converter for the settings-JSON so RichText gets displayed properly in settings.

[CMS-20315]

Fix behaviour of CMHTML data property

CMHTML data property now gets properly resolved. It is possible to retrieve a URL for a markup and download it in a separate request.

[CMS-20007]

Fix various nullability related issues

Cleared interfaces and implementing classes to properly deal with nullability annotations in Headless Server.

[CMS-19701]

[Headless] Exposing Missing SEO Properties

The missing properties for CMLinkable "htmlTitle", "keywords" and "htmlDescription" were added to the content schema.

[CMS-19690]

Fix broken handling of arrays in input values for headless persisted queries and rest endpoint

Arrays passed as input values for the persisted queries and the rest endpoint are now converted correctly and work as intended. With this it is also possible to pass arrays as parameters of queries.

[CMS-19689]

Ignore destroyed contents in LinkListAdapter

Ignore destroyed contents in LinkListAdapter to avoid returning them.

[CMS-19622]

Fix ViewBySiteFilterDataFetcher not checking DataFetcherResults

The ViewBySiteFilterDataFetcher now also properly can deal with DataFetcherResults and not just accepts them.

[CMS-19517]

Configuration of custom Solr fields

Custom Solr fields can be configured for the Headless-Server to be used in Search or QueryLists. See documentation for details.

[CMS-19277]

SchemaParser ignores type extensions for inherited types

The SchemaParser is responsible to apply the @inherit custom directive, thus apply all inherited fields throughout a type hierarchy. The inheritance did not took fields of type extensions into account. This misbehaviour was fixed.

[CMS-19073]

Headless Server sends HTTP header Strict-Transport-Security by default

Headless Server now sends by default a Strict-Transport-Security HTTP header with the default value of "max-age= 63072000; includeSubdomains". The content of the header can be configured by application properties. Please see the deployment manual for details.

[CMS-19046]

Slightly improved error message handling

For security reasons, Headless Server does not expose any textual error messages anymore, when running in live mode. This prevents any unwanted, accidental exposure of server side error messages which might leak information potentially useful for attackers. In preview mode, error messages are still part of the JSON error response.

[CMS-19045]

CoreMedia Documentation

Spring Boot Endpoint Documentation added

A description of custom Spring Boot actuator endpoints has been added to the Operations Basics handbook.

[CMS-18874]

Miscellaneous

Fix Check for Personal Data Usage

The check for usage of personal data has been fixed to work with OpenJDK 11.0.12+ by including Guava's third-party library `com.google.errorprone:error_prone_annotations`, so that annotations used by Guava are available on the classpath.

[CMS-20426]

Third-Party Update : Tomcat

Tomcat has been updated to version 9.0.54 to avoid security vulnerabilities of the previous version.

[CMS-20244]

Third-Party Update : Ant

Ant has been updated to version 1.10.11 to avoid security vulnerabilities of the previous version.

[CMS-19843]

Third-Party Update : Tomcat

Tomcat has been updated to version 9.0.48 to avoid security vulnerabilities of the previous version.

[CMS-19841]

Third-Party Update : Jetty

Jetty has been updated to version 9.4.43.v20210629 to avoid security vulnerabilities of the previous version.

[CMS-19840]

ContentHub: ContentWriteInterceptors are now correctly applied for subtypes

ContentWriteInterceptors are now correctly applied for subtypes when content is created via ContentHub.

[CMS-19830]

Third-Party Update: Netty

Netty has been updated to version 4.1.63.Final to avoid security vulnerabilities of the previous versions.

[CMS-19257]

Third-Party Update: Jetty

Jetty has been updated to version 9.4.39.v20210325 to avoid security vulnerabilities of the previous version.

[CMS-19250]

Update of MongoDB Java Driver

The thirdparty dependencies org.mongodb:mongodb-driver-legacy, org.mongodb:mongodb-driver-core and org.mongodb:bson have been updated from version 4.1.0 to 4.2.2.

[CMS-19077]

Fixed OSM Marker Initialization

Fixed issue with the OSM integration where the map marker would not appear.

[CMS-18890]

5. Upgrade Information

This chapter contains all information relevant for an upgrade to the specific release.

5.1 Upgrade to 2204.3

This section describes the necessary upgrade steps for an upgrade from the previous release. Also read [Chapter 4, Release Information \[8\]](#) to get an overview of the new features that require no immediate action on your site.

5.1.1 CoreMedia Core

Cli Tool Argument Checks are Improved

Especially the `cm bulkpublish` command and other tools have not checked their parameter and arguments. In this specific case, a folder was passed to the `bulkpublish` command without the necessary `-f` option (because other tools do it that way). However, the tool did not complain about the unknown argument and simply assumed a default value for the folder. This is now checked more closely for all tools that don't need further arguments and a usage message is printed. This change is only breaking if you have used the tools with wrong parameters. If you used them correctly, the behavior does not change.

[CMS-21210]

5.1.2 CoreMedia Content Application Engine

Fix and Deprecate Method

`TransformImageService#getTransformationString`

The method `com.coremedia.cap.transform.TransformImageService#getTransformationString(...)` has been changed to return null for non-existing blobs instead of throwing a `NullPointerException`. Further the method has been marked as deprecated for removal because it was not intended to be public API.

[CMS-21919]

5.1.3 CoreMedia Headless Server

Third-Party Update: GraphQL-Java to 19.2

The library `com.graphql-java:graphql-java` was updated from 18.1 to 19.2. It contains several bugfixes and some security relevant improvements.

The library update is potentially breaking, in case custom code implements its own `ChainedInstrumentation`. For details about the breaking change please see <https://github.com/graphql-java/graphql-java/releases/tag/v19.0> and additionally <https://github.com/graphql-java/graphql-java/releases> Otherwise, the update is considered non-breaking.

[CMS-22091]

5.2 Upgrade to 2204.2

This section describes the necessary upgrade steps for an upgrade from the previous release. Also read [Chapter 4, Release Information \[8\]](#) to get an overview of the new features that require no immediate action on your site.

5.2.1 CoreMedia Core

Log Level Change for Publication Failure

Publication failures were logged with level *WARN* in *Content Management Server* . This wrongly indicated an operational issue that would require action by the operational team, while it actually is a content issue that has to be handled by editorial staff. The log level has thus been changed to *INFO* . Feedback on publication failures in *Studio* and in command line tools remains unchanged.

If a monitoring of publication failures was established that is based on log level *WARN* in *Content Management Server* log files, this needs to be changed to filter for log level *INFO* and message text contents "*Result of publication*" and "*_Cannot_execute_*".

So, while the log message previously read

```
2022-05-20 17:52:56 [WARN] (Publication#820 on target) cap.server.publisher
[] - Result of publication '820': PublishContent[1602:3]:Cannot_execute:16
(LinkFailed)
```

it will now show as

```
2022-05-20 17:52:56 [INFO] (Publication#820 on target) cap.server.publisher
[] - Result of publication '820': PublishContent[1602:3]:Cannot_execute:16
(LinkFailed)
```

[CMS-21688]

5.2.2 CoreMedia Studio Server

Fixed Incorrect Defer Flag During Theme Import

As described in <https://documentation.coremedia.com/cmcc-11/artifacts/220/webhelp/ frontend-en/content/ThemeConfigReference.html>, you can set the "defer" flag in your theme configuration to defer the execution of a JavaScript file. Due to a bug all CMJavaScript documents generated by the theme importer that were not meant to be deferred were configured in a way that caused the following HTML output:

```
<script .... defer="false"></script>
```

This can cause problems with other flags just like "async". To avoid any problems with other flags the "defer" flag is now once again omitted if not set to "true".

(CMS-21476)

5.2.3 CoreMedia Workspace

Third-Party Update: Spring Boot and Related Dependencies

Spring Boot has been updated to version 2.6.8. Related dependencies have been updated to at least the versions used by Spring Boot. This includes updates to fix reported security issues in Spring Security and Tomcat (CVE-2022-22976, CVE-2022-22978, CVE-2022-29885). The new version of Spring Security contains a bug fix for certain configurations of password hashing (CVE-2022-22976), which may require additional upgrade steps if you have changed the configuration for password hashing in the *Content Server* or *Elastic Social*.

By default, passwords of *Content Server* builtin users and of *Elastic Social* users are stored using the BCrypt password hashing algorithm from Spring Security with a work factor of 10. Previous versions of Spring Security contained a security vulnerability (CVE-2022-22976) that made hashes less secure, if you had changed the default configuration to use work factor 31 with configuration property `cap.server.login.password-hash-algorithm` for the *Content Server* or property `elastic.social.pass`

`word-hash-algorithm` for *Elastic Social* . You don't need to take action, if you are using the default configuration or a BCrypt work factor less than 31.

With this release, the configuration of work factor of 31 is not accepted anymore. *Content Server* and *Elastic Social* applications will fail to start, if such a work factor is configured. Note, that it would be an unreasonable choice anyway, because it would require multiple days to compute a single hash. If you had configured work factor 31, then you must change the configuration in the mentioned configuration properties, for example to the default work factor of 10, or a slightly higher one. The passwords of all users that were stored with work factor 31 are still working but should be changed to store them more securely. For details, see also the description of the CVE at <https://tanzu.vmware.com/security/cve-2022-22976> .

The following libraries have been updated:

- Hibernate 5.6.9.Final
- Jackson 2.13.3
- Micrometer 1.8.6
- MySQL 8.0.29
- PostgreSQL 42.3.6
- Spring Boot 2.6.8
- Spring Security 5.6.5
- Tomcat 9.0.64

If you use these libraries in project code, please check their respective release notes for changes and upgrade information. No changes were necessary in the CoreMedia Blueprint for these updates.

[CMS-21777]

5.2.4 CoreMedia Headless Server

Picture Local Setting 'disableCropping' now Supported by Link Building

The flag `disableCropping` in the local settings of a picture was not respected by the link building for the GraphQL property `uriTemplate` . This misbehaviour was fixed. Also, the delivery will now validate this flag. This means that uri with disabled cropping won't work when request the image with a cropName and width. As a con-

sequence, this bugfix is considered breaking, as the validation may break previously working delivery image URLs!

For version 2107 and 2201 the validation for the delivery can be disabled. For version 2204 validation cannot be disabled!

[CMS-21765]

5.2.5 Miscellaneous

Fixed Prometheus Cache Metrics

Spring Boot actuator endpoint `/metrics` provides several metrics on CoreMedia components [see *Operations Basics Manual* , section *Basics of Operation | Actuator Endpoints | Metrics Endpoint*]. In addition to that, actuator endpoint `/prometheus` publishes this data for scraping by monitoring application *Prometheus* . However, cache metrics were incomplete for some CoreMedia components when using this endpoint. With this fix, the complete set of cache information can now be retrieved from it.

CoreMedia cache meter names in Spring Boot actuator endpoints `/metrics` or `/prometheus` have been changed:

- `cache.capacity` -> `coremedia.cache.capacity`
- `cache.evaluation.duration` -> `coremedia.cache.evaluation.duration`
- `cache.eviction.duration` -> `coremedia.cache.eviction.duration`
- `cache.evictions` -> `coremedia.cache.evictions`
- `cache.gets` -> `coremedia.cache.gets`
- `cache.level` -> `coremedia.cache.level`
- `cache.puts` -> `coremedia.cache.puts`
- `cache.removals` -> `coremedia.cache.removals`
- `cache.size` -> `coremedia.cache.size`
- `cache.updates` -> `coremedia.cache.updates`

Tag `cache` is not used on these meters any longer.

Meter names `cache.gets` , `cache.puts` , `cache.size` , and `cache.evictions` may still be present for other caches (depending on the component).

If you used cache metrics before, you may need to adjust your monitoring application's configuration according to the list above to comply with the new meter names for CoreMedia caches. See *Operations Basics Manual* , section *Basics of Operation | Actuator Endpoints | Metrics Endpoint* for details on metrics parameters.

[CMS-21612]

5.3 Upgrade to 2204.1

This section describes the necessary upgrade steps for an upgrade from the previous release. Also read [Chapter 4, Release Information \[8\]](#) to get an overview of the new features that require no immediate action on your site.

5.3.1 CoreMedia Core

Class "TransformationService" Cleaned Up

Class `TransformationService` has been refactored to delegate image transformations to a new method in `TransformImageService`. As a consequence, obsoleted references to `BlobTransformer` and `NamedTransformBeanBlobTransformer` have been removed. `TransformationService`'s constructor as well as the corresponding Spring bean configuration in `TransformationServiceConfiguration` have been adjusted accordingly.

[CMS-21491]

New Method to Transform Image by "Image Transformation String" Directly

A new method `transformWithDimensions(Content content, String property, String operations):Optional<Blob>` was added to interface `TransformImageService` and its implementing class. With this new method, an image transformation can be requested via an Image Transformation String directly (see <https://documentation.coremedia.com/cmcc-11/artifacts/220/webhelp/cas-developer-en/content/imageTransformationAPI.html>).

Existing transformation methods remain unchanged.

[CMS-21481]

Unified API: `ContentRepository.createContentBuilder()` Available

A new way of creating contents within the Unified API is available:

- `ContentRepository.createContentBuilder()`

It combines several aspects of content creation such as `ContentRepository.createChild(...)` and `ContentType.createByTemplate(...)`.

The most simple approach of creating a content is:

```
repository.createContentBuilder()  
    .type("CMArticle")  
    .name("My Article")  
    .create();
```

which will create an article directly at root-folder.

For more details, consult the corresponding JavaDoc in Common API of

- `com.coremedia.cap.content.create.ContentBuilder`

Upgrade Notes:

The pattern has been introduced to some production and test files in CoreMedia Blueprint. While, generally, no upgrade steps are required, you may experience merge conflicts, when you adapted corresponding code sections.

If any conflicts arise, it is safe keeping your changes in favor over the changes introduced with this update. All former methods will work in the same way.

Affected production files:

- `DefaultTaxonomy.java` in `apps/studio-server/blueprint/modules/shared/taxonomies/src/main/java/com/coremedia/blueprint/taxonomies/strategy`
- `SettingsUploadInterceptor.java` in `apps/studio-server/blueprint/modules/extensions/validators/validators-studio-lib/src/main/java/com/coremedia/blueprint/studio/rest/intercept`
- `TopicPagesResource.java` in `modules/extensions/custom-topic-pages/custom-topic-pages-rest-extend`


```
sion/src/main/java/com/coremedia/blueprint/studio/top
icpages/rest
```

- `UpdateAssetMetadataWriteInterceptorTest.java` in `modules/extensions/am/am-studio-component/src/test/java/com/coremedia/blueprint/assets/studio/intercept`
- `WordUploadInterceptor.java` in `apps/studio-server/blueprint/modules/extensions/validators/validators-studio-lib/src/main/java/com/coremedia/blueprint/studio/rest/intercept`

Affected test files:

- `CMQueryListImplPaginationTest.java` in `shared/middle/blueprint/modules/cae/contentbeans/src/test/java/com/coremedia/blueprint/cae/contentbeans`
- `LocalizationServiceTest.java` in `shared/middle/blueprint/modules/shared/content-services/src/test/java/com/coremedia/blueprint/localization`

[CMS-18001]

Runlevel Health Indicator Signals Wrong State in Runlevel MAINTENANCE

When the Content Server is switched to runlevel MAINTENANCE either during startup or manually, the `health/runlevel` endpoint signals the server is UP. This makes it impossible to use the indicator for a readiness probe in Kubernetes, because the pod will be added to the services endpoint too early and UAPI clients that listen on a services availability would fail during the creation of the repository connection.

This change fixes the health state in the MAINTENANCE runlevel to down and includes the indicator in the `health/readiness` endpoint.

The content-server can therefore be in one of the following states:

- `runlevel=OFFLINE`, `Liveness State=BROKEN`, `Readiness State=REFUSING_TRAFFIC`
- `runlevel=MAINTENANCE`, `Liveness State=CORRECT`, `Readiness State=REFUSING_TRAFFIC`
- `runlevel=ADMINISTRATION`, `Liveness State=CORRECT`, `Readiness State=REFUSING_TRAFFIC`

- runlevel=ONLINE, Liveness State=CORRECT, Readiness State=ACCEPTING_TRAFFIC

If your health check logic depended on MAINTENANCE level being a 200 HTTP result, disable the runlevel endpoint by setting

```
management.health.runlevel.enabled=false  
management.endpoint.health.group.readiness.include=readinessState
```

Please be reminded, that using the health actuator to automatically trigger a redeployment of a pod / container is not advised and can be dangerous due to health indicator that monitor connections to remote systems. This can cause chain reactions taking multiple services down. Instead use the `host:8081/actuator/health/liveness` endpoint instead.

[CMS-21355]

Changed SitesService Setup in cmd Tools

The `sitesService` bean is injected from `commandline-tools-context.xml` now instead of being registered programmatically. Now obsolete methods `registerSitesService` and `hasSingletonBeanOfType` in `com.coremedia.cmdline.SpringHelper` have been removed.

[CMS-21279]

5.3.2 CoreMedia Studio Client

Moved thumbnailService and Helper Classes to cap-base-models

This note covers changes to private API. Although private API is not subject to release notes and it can be changed without further notice we detected that this API was leaked in the blueprint so the code might have been copied or customized that is why we declared this change as possibly breaking.

The following non-Public API modules (which were used in the blueprint) are now deprecated:

- @coremedia/studio-client.ext.cap-base-components/thumbnails/IThumbnailService
- @coremedia/studio-client.ext.cap-base-components/thumbnails/ThumbnailResolver

- @coremedia/studio-client.ext.cap-base-components/thumbnails/ThumbnailResolverFactory
- @coremedia/studio-client.ext.cap-base-components/thumbnails/ThumbnailResolverImpl

Please make sure to read the deprecation notes and adjust your usages before the modules are removed.

For the sake of completeness: The following non-Public modules (not used in the blueprint) were moved:

- @coremedia/studio-client.ext.cap-base-components/thumbnails/RemoteImageTransformationConfig -> @coremedia/studio-client.cap-base-models/thumbnails/RemoteImageTransformationConfig
- @coremedia/studio-client.ext.cap-base-components/thumbnails/Thumbnail -> @coremedia/studio-client.cap-base-models/thumbnails/Thumbnail
- @coremedia/studio-client.ext.cap-base-components/thumbnails/ThumbnailRenderer -> @coremedia/studio-client.cap-base-models/thumbnails/ThumbnailRenderer
- @coremedia/studio-client.ext.cap-base-components/thumbnails/ThumbnailRendererBase -> @coremedia/studio-client.cap-base-models/thumbnails/ThumbnailRendererBase
- @coremedia/studio-client.ext.cap-base-components/thumbnails/thumbnailService -> @coremedia/studio-client.cap-base-models/thumbnails/thumbnailService
- @coremedia/studio-client.ext.cap-base-components/thumbnails/ThumbnailServiceImpl -> @coremedia/studio-client.cap-base-models/thumbnails/ThumbnailServiceImpl
- @coremedia/studio-client.ext.cap-base-components/thumbnails/ThumbnailUriServiceImpl -> @coremedia/studio-client.cap-base-models/thumbnails/ThumbnailUriServiceImpl

In case you made use of this private API please adjust your imports accordingly.

[CMS-22223]

Third-Party Update: rxjs from 6.x to 7.x

We now use the latest major version of rxjs in the studio-client workspace. In case you made any use of it in one of your packages make sure to upgrade rxjs as well. The changes can be seen here: <https://rxjs.dev/deprecations/breaking-changes>

[CMS-22220]

Added Support for Multiple Bookmarks

Inside the library, multiple content items can be bookmarked now. Additional properties have been added to `_Editor_properties.ts_` for this:

```
Bookmark_button_add_multi_text: "Add Bookmarks",  
Bookmark_button_add_multi_tooltip: "Bookmark the selected content items",  
Bookmark_button_remove_multi_tooltip: "Remove bookmarks",  
Bookmark_button_remove_multi_text: "Remove Bookmarks",
```

[CMS-21454]

Studio Client Unified API: Improved TypeScript Types

The part of Studio Client API that corresponds to CMCC's Unified API in Java (content, workflow, users) has been overhauled to provide more precise TypeScript types. Studio Client TypeScript code has been migrated from `ActionScript`, where it was not possible to describe callback function signatures and certain complex types (unions, intersections, generic types / type parameters). With TypeScript, now it is. To provide an improved formal description of the API, all vague types like `AnyFunction` or `any` have been replaced by more specific types, mostly already given in the textual documentation. This affects the packages `@coremedia/studio-client.client-core` and `@coremedia/studio-client.cap-rest-client`. Formally, changing the API to be more specific is a breaking change. But the required callback signatures, array element types and union types have already been documented as text, so if your code adheres these constraints, it is not going to break. Otherwise, only the TypeScript types of your code might have to be adapted, or the new, more precise types may even find hidden bugs in client code that need to be fixed.

[CMS-21089]

Fixed babel Configuration in studio-client

We fixed our internal babel configuration for the `studio-client` so code is no longer transpiled using the so called "loose" mode. This will cause that the resulting JavaScript code is now more precise regarding ES6 standards than before.

While in most cases this should not lead to any problems when upgrading to the new version in some corner cases the fixed behavior might interfere with some workarounds you might have introduced to circumvent the fact that the code was not exactly transpiled

down before the change. For code migrated from ActionScript that was not adjusted in any way we do not expect any problems at all.

An addition to this it is now possible to use new ES6 language features like "optional chaining" and the "nullish coalescing operator" without Sencha CMD complaining about syntax errors.

[CMS-21144]

5.3.3 CoreMedia Studio Server

Enhancement: Custom Validation for Built-In Workflows

Custom start validators for built-in workflows (translation, synchronization, simple publication, and two-step publication) can now be configured via properties.

Each workflow receives a property on its own. In case you modified start validators of any of the built-in workflows, it is recommended you adapt to the new customization approach as described in *Studio Developer Manual*, section *Customizing Validation of Built-In Workflows*.

Bean `defaultPublicationValidators`, defined in `PublicationWorkflowValidationConfiguration` does not exist any longer. This change is necessary to support the new customization approach. In case this bean is modified in your project, the new customization approach must be applied as described in *Studio Developer Manual*, section *Customizing Validation of Built-In Workflows*. You must then apply your change in properties for both simple publication and two-step publication workflows. If the bean has previously been used in validation of custom workflows, the list of product-wise defined publication validators can be retrieved through method `getWorkflowStartValidators()` of bean `simplePublicationWorkflowValidators`.

[CMS-19126]

5.3.4 CoreMedia Workspace

Third-Party Update: Spring Boot and Related Dependencies

In order to benefit from the improvements of the latest version of Spring Boot and the Spring framework and to prepare for making use of the new features various 3rd party libraries have been updated.

- AspectJ 1.9.8
- AssertJ 3.21.0
- Byte Buddy 1.11.22
- Hibernate 5.6.8.Final
- Hibernate Validator 6.2.3.Final
- Jackson 2.13.2.1
- Janino 3.1.7
- JBoss Logging 3.4.3.Final
- Jetty 9.4.46.v20220331
- JUnit 5.8.2
- Logback 1.2.11
- Maven Jar Plugin 3.2.2
- Micrometer 1.8.5
- Mockito 4.0.0
- Netty 4.1.76.Final
- Postgres Driver 42.3.4
- Protobuf 3.19.4
- Spring Boot 2.6.7
- Spring Data 2021.1.4
- Spring Framework 5.3.20
- Spring Security 5.6.3
- Thymeleaf 3.0.15.RELEASE
- Tomcat 9.0.62

For more details on the changes in the Spring Boot 2.6 release, see <https://github.com/spring-projects/spring-boot/wiki/Spring-Boot-2.6-Release-Notes>.

Upgrade Steps

- With Mockito 4.0.0 all deprecated API was removed. Usages of the API in custom tests have to be replaced by its more recent counterparts.
- With Spring Boot 2.6.x the application automatically checks for circular bean references. We resolved all circular references mostly by using `@Lazy` to break the cycle.
- With Spring Boot 2.6.x the path matching strategy was changed from the `Ant Path Matcher` to the `Path Pattern Parser`. You can change the default by setting the configuration to `"spring.mvc.pathmatch.matching-strategy=ant_path_matcher"` in custom applications.
- Studio Server Spring Security Customization: Due to migration to Java Configuration the abstract bean `abstractSpringSecurityCapUserFinder` has been removed. See section "Single Sign On Integration" in the Studio Developer Manual for details.
- There might be more changes that affect your application. Refer to the official release notes mentioned above if you encounter any issues.

[CMS-20445]

5.3.5 CoreMedia Deployment

Refactorings of the Traefik Configuration for the Developer docker-compose Setup

The Traefik configuration for the docker-compose developer setup has been refactored.

- Replace static file config with environment variables. This refactoring has been done to allow other docker-compose files to alter the static traefik configuration without duplicating the whole static configuration file. In our internal setup, we use this to enable access logging.
- Refactor the traefik dashboard routing from docker config to file. This was the last router configured by docker labels. Therefore the docker configuration provider could be removed from the setup. Also all the `traefik.enabled: true` labels have been removed as they were obsolete.

- The endpoints for http/https had to be renamed to names without hyphens to conform to a naming convention, that allows the endpoints to be configured using environment variables.
- Traefik has been updated to 2.6.3.
- Traefik Pilot integration in the dashboard has been disabled.

The change is breaking only in the sense of the docker-compose development setup now uses endpoints with different names and added services with ingress need to change the configuration accordingly. The endpoints were renamed:

- `web-access-insecure` to `webaccessinsecure`
- `web-access-secure` to `webaccesssecure`

With the previous setup a customer could also have added services using docker labels for the traefik configuration, this also no longer works and should be refactored using config files.

[CMS-21441]

Fixed Directory Ownership in Container Images

The ownership of the `/coremedia` directory was root but should be coremedia.

[CMS-21079]

5.3.6 CoreMedia Connector for SAP Commerce Cloud

Deleted Demo Content Site Apparel US

Demo content for site Apparel US was deleted because SAP Commerce does not provide a US site by default. The Apparel UK and Apparel DE still remain in demo content as example sites for the SAP Commerce integration.

[CMS-21432]

5.3.7 CoreMedia Headless Server

New Search Adapter with ExtensionPoint

The former `SearchServiceAdapter` and corresponding classes were replaced by the new `SearchAdapter`, featuring a SPI architecture and offering `SearchServiceProvider` as an `ExtensionPoint`. The default implementation in `DefaultSearchServiceProvider` implements the same behaviour as the former `SearchServiceAdapter`.

This change is **considered breaking**, because the new `SearchAdapter` is now used in the `search-schema` (implementing the former behaviour). If the now outdated `SearchServiceAdapter` was not extended or changed, this change behaves as **nonbreaking**.

In case, `SearchServiceAdapter` was extended or changed, `SearchServiceAdapter` is just deprecated and still part of `HeadlessServer`. If necessary, the old adapter may still be used the same way as before simply by changing the search-schema:

```
// from
search(
    ...
): SearchResult @fetch(from: "@searchServiceAdapter.to().search(#query,
#offset, #limit, #docTypes, #sortFields, #siteId, #includeSubTypes,
#customFilterQueries)")

// to
search(
    ...
): SearchResult @fetch(from: "@legacySearchServiceAdapter.to().search(#query,
#offset, #limit, #docTypes, #sortFields, #siteId, #includeSubTypes,
#customFilterQueries)")
```

[CMS-21256]

FilterQueryArg now Part of Plugin Support Library

In order to enable customizations of the search via plugin extension points, it was necessary to move input type mode `FilterQueryArg` to plugin support. As a consequence the package has changed. Otherwise this change is nonbreaking.

[CMS-21076]

Expose SolrQueryBuilder as Beans for Plugins

In order to enable customizations of the search via plugin extension points, it was necessary to expose the so called `SolrQueryBuilder` as beans for plugins. As a consequence, `SolrQueryBuilder` had to be refactored to prevent circular bean dependencies with other search related extension points.

Customizations which extend or use instances of `SolrQueryBuilder` must be adapted, as one constructor parameter and one getter was removed. The 'filterQueryDefinitionMap', previously provided by the now removed getter, is now provided as a regular SpringBean. In case the 'filterQueryDefinitionMap' was consumed, it has to be provided via Spring injection instead.

If neither `SolrQueryBuilder` was adapted, nor its getter `getFilterQueryDefinitionMap()` was used, this change is considered **nonbreaking** !

For details about the new exposed 'beans for plugins', please see the Headless Manual.

[CMS-21063]

MediaController Validates URL Parameters

The `MediaController` now validates the URL parameters 'id', 'propertyName', 'hash', 'cropName', 'width' and 'filename'. The missing validation was added for security reasons to prevent DOS attacks.

This change is considered breaking due to the validation, which now might deny requests which previously worked.

Valid URLs for media objects delivered by the `MediaController` are 'published' via the GraphQL schema in the properties 'uriTemplate' and 'fullyQualifiedUrl'

[CMS-21317]

5.3.8 Miscellaneous

Blob Upload Robustness

Sometimes, CMCC client threads got stuck due to broken BLOB upload HTTP connections.

Now the upload is controlled by the same timeout as the upload operation on the server side (plus a little overhead), and the threads are released if the upload exceeds the timeout. Both, Content Servers and clients, must be restarted to make the fix effective.

[CMS-19390]

5.4 Upgrade to 2201.1

This section describes the necessary upgrade steps for an upgrade from the previous release. Also read [Chapter 4, Release Information \[8\]](#) to get an overview of the new features that require no immediate action on your site.

5.4.1 CoreMedia Core

Reduced Overhead for Background Task CleanInTranslation

You may experience the Workflow Server's Background Task `CleanInTranslation` to cause considerable load on your CPU. We have addressed this issue with several adaptations. Some of them might be applied manually to existing systems already (for details see the end of this section). In addition to the updates below, `CleanInTranslation` got updated internally, providing an additional performance gain.

DB Index for `Resources (nMergeVersions_)`

On update, a new index will be created for the Content Management Server:

```
create index nmergeversionsIndex on Resources (nMergeVersions_);
```

There's no problem for the update, if you have created the exact same index manually before.

Configure Scheduling

`BlueprintWorkflowServerAutoConfiguration` got extended by properties to control the scheduling behavior of `CleanInTranslation`. The following shows an example configuration using the defaults:

```
workflow.blueprint.clean-in-translation.initial-delay=10s  
workflow.blueprint.clean-in-translation.fixed-delay=15m
```

Thus, the initial delay is 10 seconds (same as before), and the task is repeated every 15 minutes (before: every 5 seconds).

Note, that while you may provide the delays without units (which defaults to milliseconds then), it is recommended to clearly state the desired unit. For details on configuring duration properties, see [Converting Durations](#) in Spring Boot Reference Guide.

For more details, see *Blueprint Developer Manual* in the subsection *Configuration and Customization* of the section *Translation Workflow*.

At workflow server startup, you will see a message similar to this to inform you about scheduling:

```
Scheduling CleanInTranslation: startTime: 2021-12-01T08:00:00Z, initialDelay: PT10S, fixedDelay: PT15M
```

Optimize Scheduling

Optimizing scheduling means to deal with the trade-off of having more correct reporting of *in-translation* state for abnormally ended workflows (aborted or escalated) versus lowering load on the workflow-server.

If you enable `INFO` logging for `com.coremedia.translate.workflow.CleanInTranslation`, the following status messages may provide some guidance for tuning scheduling:

```
CleanInTranslation - Merge-Version Cleanup: Done within 2.655 s, cleaned up merge versions: 1.
```

Note, that cleaning up 0 (zero) merge versions should be the normal state on a system, where all translation workflows end normally.

Workaround for Previous Versions

- Add the index to the content management server's database, as stated above.
- Update the `@Scheduled` parameters especially for `fixedDelay` in Blueprint class `BlueprintWorkflowServerAutoConfiguration`

[CMS-20366]

Fixing Possibly False-Negative In-Translation State

The scheduled job `com.coremedia.translate.workflow.CleanInTranslation` is, roughly speaking, responsible for garbage collecting meanwhile invalid "in-translation" states of contents due to aborted or escalated translation workflows.

Prior to this fix, it may have made false-positive detection of states (so-called merge-versions) to clean up. As a result, you may have observed inconsistencies in translation state widget, for example.

The fix extends `CleanInTranslation` by an addition parameter `confidenceThreshold`. Simply speaking, it tells `CleanInTranslation` how often a state should be reported as possibly dirty, before it is going to clean up.

A threshold of 0 [zero] triggers the old behavior, which is immediate clean-up. The recommended threshold is 1 [one]. This is also the default which comes with Blueprint configuration now.

Upgrade Notes:

Unless you modified classes in Workflow Server App (`BlueprintWorkflowServerConfigurationProperties` and/or `BlueprintWorkflowServerAutoConfiguration`), changes should be automatically merged.

In `BlueprintWorkflowServerConfigurationProperties` a new property `workflow.blueprint.clean-in-translation.confidence-threshold` has been introduced, which, as stated above, defaults to 1.

The corresponding `CleanInTranslation` bean is instantiated with this new threshold.

For details, see section *Configuration and Customization* in Blueprint Developer Manual's description of the Translation Workflow.

[CMS-16439]

5.4.2 CoreMedia Content Application Engine

New method

`TransformImageService#getTransformationString`

`com.coremedia.cap.transform.TransformImageService` has been extended by a new method: `getTransformationString(...)`. The method was introduced to enable future use cases of this class. You can use it to compute the transformation string for a given content, property, variant name, breakpoint width and height, and extension. If you have implemented this interface in a custom class, please make sure to implement this method as well. It is not used by default in the Blueprint, so you can theoretically leave the implementation empty.

[CMS-20410]

The preview CAE now runs with standalone mode disabled

The property `delivery.standalone` is now disabled by default in the preview CAE. As a result the preview CAE uses the same URLs as the live CAE and the default proxy configuration no longer requires `/blueprint/servlet` to be accessible when running behind a web proxy.

[CMS-20501]

Static Files of External Preview moved to /static folder

In order to simplify proxy rules to split requests to static files and dynamic servlets, the static files of the external preview have been moved to a static folder and are now served at

```
<context root>/static/externalpreview
```

[CMS-20471]

Freemarker function `bp.truncateText` always returns un-encoded text

The Freemarker function `bp.truncateText()` lead to double encoded text, when used with markup, because the markup is first encoded by the function and secondly by Freemarker itself. Now, the encoding is left to do by Freemarker and the function returns un-encoded text for markup input as well as for strings. An implemented work-around with `"?no_esc"` is no longer needed and should be removed.

[CMS-20431]

5.4.3 CoreMedia Studio Client

ToastManager has been changed to toastService

The access to the *ToastManager* has been changed. It has been replaced by the global service *toastService*.

[CMS-20963]

New Rule REST API

With the introduction of the dialog "Folder Properties" a new REST API has been introduced for managing group and content permission via rules. *Content.ts* and *Group.ts* support the method *getRules* now. It returns a remote bean which contains a list of rule remote beans. These ones are displayed either in the Folder Properties dialog or on the rule tab of a user managers group tab.

The former implementation of rule customization based on plain JSON is gone. Therefore this ticket has been marked as breaking change.

[CMS-20732]

Strict Signature of Content Initializers

The Content Initializer API has been stated more formally precise by specifying the initializer function signature as a TypeScript type `(content: Content) => void`. This concerns the following methods in `IEditorContext` [@coremedia/studio-client.main.editor-components/sdk/IEditorContext] and `IContentTypeContext` [@coremedia/studio-client.cap-base-models/content/IContentTypeContext]:

- `registerContentInitializer()`
- `extendContentInitializer()`
- `getContentInitializer()`
- `lookupContentInitializer()`

If adhering to the documentation of these methods, your client code should already have the required function signature. If not, it should be easy to adapt by changing your function signature accordingly or by using an adapter arrow function like this:


```
(content: Content) => { this.myContentInitializer(content) }
```

[CMS-20652]

Third-Party Update: CKEditor 4 to 4.17.1

CKEditor 4 has been updated to 4.17.1. It is included in `@jangaroo/ckeditor4`, version 1.1.0.

For details, see recent release notes:

- ckeditor.com/cke4/release/CKEditor-4.16.2
- ckeditor.com/cke4/release/CKEditor-4.17.0
- ckeditor.com/cke4/release/CKEditor-4.17.1

This update comes with XSS vulnerability fixes rated as important by CKEditor.

While in CoreMedia Blueprint, no adaptations had to be applied using the new CKEditor 4 version, CKEditor did some API changes with release of version 4.17.0 which may affect your adaptations.

Note, that as of 4.17.0 the Flash Plugin [ckeditor.com/cke4/addon/flash] has been removed from CKEditor 4. If you depend on this plugin, consider carefully, if you may skip the update of `@jangaroo/ckeditor4`.

[CMS-20580]

Studio Client Resource Bundle Changes

In the process of pulling features out of the editor-components module in order to make them available for the new Workflow app, many changes to resource bundles had to be made (these changes are in addition to the changes that happened due to the new content type localization mechanism). As the list is too extensive for a release note, we refer to [this link to a Coremedia Help Center article](#).

[CMS-18728]

Master Version Texts Improved

The description texts above the master version property field have been improved. Existing texts have been removed from the *Editor.properties* resource bundle. For this reason, the ticket is marked as breaking.

[CMS-20989]

5.4.4 CoreMedia Workspace

Use new Maven Plugin to Handle CoreMedia Plugins

CoreMedia plugins are now synchronized by means of the `coremedia-plugins-maven-plugin` which handles adding, removing and downloading the respective plugins. The Groovy scripts in the `<workspace>-app/pom.xml` files (e.g. `cae/spring-boot/cae-live-app/pom.xml`) have been replaced with the `coremedia-plugins-maven-plugin` which supports caching of downloaded artifacts and respects Maven's proxy settings.

[CMS-20775]

SFMC Extensions Extracted from Blueprint

The extensions `sfmc`, `sfmc-p13n` and `sfmc-es-p13n` have been removed from Blueprint. The extensions for your version can be requested from the CoreMedia Support Team.

[CMS-20575]

5.4.5 CoreMedia Connector for Salesforce Commerce Cloud

Link Building for Augmented Pages [commerce-led]

For Augmented Pages (CMExternalPage) the external page ID and the external URI path can be set. If configured both, both values will now be rendered into the Json/VTL include template of the fragment response. Make sure your link rendering template on the commerce side can handle both parameters.

[CMS-19581]

5.4.6 CoreMedia Headless Server

Refactored Scalar FilterQueryArg to input type CustomFilterQuery

The former scalar `FilterQueryArg`, which was used to define the custom filter queries in a search query, was refactored to a GraphQL input type of the same name. In contrast to complex scalars, input types offer full support of auto completion and documentation while formulating a GraphQL query.

The refactoring to the GraphQL input type is **non-breaking**, so that client applications don't have to change their code.

This change took advantage of CMS-20327 in the manner, that a Spring `GenericConverter` was used to convert the `FilterQueryArg` into the same semantics as the previous GraphQL Scalar. It was however necessary, to change the Java interface `PagingAware` slightly and in turn also the `QueryListAdapter`, which implements that interface. That's why this change is considered **breaking**. In the case, that the `QueryListAdapter` was adapted or the interface `PagingAware` was used, the code has to be slightly changed.

[CMS-20620]

Added CacheControl support to Headless Server

Headless Server now supports sending CacheControl headers for GET requests. Defaults can be configured using the `caas.cache-control.*` properties. The value of the the max-age directive is the minimum of the configured max-age value for the given request URL and the values of the validFrom/validTo properties of the requested contents. If no cache control configuration exists and the content does not contain a value for validFrom/validTo then no CacheControl header is sent. A negative max-age value indicates that no CacheControl header should be sent even in the presence of configured validFrom/validTo dates.

Headless Server can be configured to expose the upcoming change dates of contents (according to validFrom/validTo settings) as part of the JSON response in addition to the CacheControl header. The upcoming change dates are added to the JSON response as `changeDates` list to the `extensions` property if `caas.expose-extensions=true`.

[CMS-19110]

5.4.7 Miscellaneous

Removed Usage of PowerMock Java Test Library

Previous releases used the PowerMock library in a small number of Java unit tests to mock static methods and object construction. These usages have now been replaced with new functionality of the Mockito testing library. PowerMock isn't used anymore.

This was changed to prevent future incompatibilities between Mockito and PowerMock. We recommend to replace custom usages of PowerMock as well, if possible.

If you still want to use PowerMock in custom classes, you have to manage the Maven dependency and version yourself. PowerMock is now also listed as banned dependency by default. See the "Blueprint Developer Manual", section "Blueprint Workspace for Developers | Concepts and Architecture | Structure of the Workspace | Enforcer" for information how to customize banned dependencies.

[CMS-20448]

New ExtensionPoint CustomFilterQuery

The new release offers the new `ExtensionPoint CustomFilterQuery` in order to provide custom filter queries by a plugin.

The introduction of this new extension point made it necessary to move the existing code base for the search into Blueprint Base. Thus, this is considered a breaking change!

However, if search related code in Blueprint was not adapted in any way, this change does not affect an update.

If the former search extension was adapted, the code changes should be moved into an own extension or a plugin depending on the grade of customization. Changes in the `search-schema` and custom filter queries can easily be moved into a plugin. This has the advantage, that future changes within headless-server should have lesser impact regarding changes in the public API, because ExtensionPoints are considered more stable and version independent.

Extensions which depend on the former search extension must now depend on `headless-server` instead.

Self implemented custom filter queries using the former `HeadlessSearchConfiguration` must be provided now as a Spring bean, using the new `ExtensionPoint CustomFilterQuery`. Examples can be found in the updated headless manual.

If the former `search-schema` was changed, the now provided `search-schema` can be disabled, setting configuration parameter `'caas.search.loadSearchSchema=false'`. Instead provide your own schema using the plugin mechanism for resource files or provide the schema file using an extension.

If Java code of the former search extension was replaced, e.g. SpringBeans, you have to use Spring annotations like `@Customize` to override the provided SpringBeans. Alternatively it is possible to disable the headless-search completely and provide an own implementation of the search as an extension.

[CMS-20367]

Third-Party Update: mysql-connector-java

The mysql-connector-java Java library has been updated in `shared/common/modules/shared/database-drivers/pom.xml` to version 8.0.28 to avoid security vulnerability CVE-2022-21363 of the previous version. Note, that the updated version of mysql-connector-java also dropped support for deprecated TLSv1 and TLSv1.1 connec-

tion protocols and requires using more-secure TLSv1.2 or TLSv1.3 protocols, which have to be supported by the used MySQL server.

For details, see the release notes of the mysql-connector-java driver in <https://dev.mysql.com/doc/relnotes/connector-j/8.0/en/news-8-0-28.html> .

[CMS-20962]

5.5 Upgrade to 2110.1

This section describes the necessary upgrade steps for an upgrade from the previous release. Also read [Chapter 4, Release Information \[8\]](#) to get an overview of the new features that require no immediate action on your site.

5.5.1 CoreMedia Core

Removal: Deprecated Version of XliffExporter

The deprecated interface `com.coremedia.translate.xliff.exporter.XliffExporter` (cap-xliff) and its related interfaces and classes have been removed.

Use the refactored, more flexible `com.coremedia.cap.translate.xliff.XliffExporter` instead.

Changes in detail:

The following classes/interfaces have been removed:

- `com.coremedia.translate.xliff.exporter.XliffExporter`

Use `com.coremedia.cap.translate.xliff.XliffExporter` instead.

- Replaced by `com.coremedia.translate.item.TranslateItem`:
 - `com.coremedia.translate.xliff.exporter.XliffExportElementCollectionBuilder`
 - `com.coremedia.translate.xliff.exporter.XliffExportElementBuilder`
 - `com.coremedia.translate.xliff.exporter.XliffExportElement`
- now unused `com.coremedia.translate.xliff.exporter.CapXliffExportException`
- now unused `com.coremedia.translate.xliff.CapXliffException`

Ease Upgrade

Some classes and interfaces have been extended, to provide the same behavior as the now removed, previously deprecated `XliffExporter`.

The interface `ContentToTranslateItemTransformer`, which is required as intermediate step to transform contents (master and derived) to translate-items accepted by the current `XliffExporter`, has been extended by using a default locale mapper, by introducing the following default methods:

- `getDefaultLocaleMapper()`, and
- `transform(masterSources, derivedTargets, strategy)`

This default is the same as it has been used by the removed `XliffExporter`.

The `XliffExportOptions` have been extended to provide a way to specify a default comment to apply to all translate items, which don't have a comment yet (added to XLIFF as `<note>`). See `XliffExportOptions.DefaultCommentOption` and its usages for details. This simulates the same behavior as the removed `XliffExporter`, where you could hand over a comment on XLIFF export – and which also got applied to all translate items, and thus to all XLIFF `<file>` entries.

And to ease, having a base set of options to modify for example by a new default comment, `XliffExportOptions` now provides a method `builder()` to create a derived set of options from the original one.

Typical Replacement

A typical replacement for the old `XliffExporter` is similar to this:

```
XliffExportOptions xliiffExportOptions = XliffExportOptions.xliffExportOptions()
    .option(XliffExportOptions.TargetOption.TARGET_EMPTY)
    .option(XliffExportOptions.DefaultCommentOption.of("default
comment"))
    .build();
List<TranslateItem> items = contentToTranslateItemTransformer
    .transform(masterContentObjects, derivedContents,
    TransformStrategy.ITEM_PER_TARGET)
    .collect(toList());

Xliff xliiff = xliiffExporter.exportXliff(items, xliiffExportOptions);
```

For details, see section *XLIFF Integration in Blueprint Developer Manual* and corresponding JavaDoc.

Replacing `setXliffExportElementFilter`

The former `XliffExporter` provided a method for filtering contents, which shall not be added to the XLIFF: `setXliffExportElementFilter`.

In general, such filtering should happen when creating the `TranslateItem` instances (this is a so-called pre-processing stage). For details, see *XLIFF Export in Blueprint Developer Manual* (subsection of *XLIFF Integration*).

Nevertheless, in contexts, where you cannot directly influence the creation of translate items, a fallback exists for the new `XliffExporter : setTranslateItemFilter`. This applies for example to the default XLIFF download as provided in CoreMedia Studio. If you require more details about the involved contents, you may want to replace the `ContentToTranslateItemTransformer` instead, which defaults to `DefaultContentToTranslateItemTransformer`.

If it is about not repeating translation for unchanged properties, you may want to set `translate.item.include-unchanged-properties` to `false`, which will be taken into account when generating the translate-items.

[CMS-20068]

Removal: Deprecated two-argument constructor of `DefaultContentToTranslateItemTransformer`

The deprecated two-argument constructor of `com.coremedia.translate.item.DefaultContentToTranslateItemTransformer` (cap-`translate-item`) has been removed:

- `DefaultContentToTranslateItemTransformer (Function, Collection)`

Instead, use the three-argument constructor:

- `DefaultContentToTranslateItemTransformer (Function, Collection, boolean)`

where the third parameter is `includeUnchangedProperties`. To get the same behavior as the old two-argument constructor, set `includeUnchangedProperties` to `true`.

[CMS-20067]

Removal: Deprecated Interface Method `ContentObjectSiteAspect.getIgnoreUpdates`

The deprecated interface method `ContentObjectSiteAspect.getIgnoreUpdates()` has been removed.

Use `ContentObjectSiteAspect.isIgnoreUpdates()`, which provides an easier to understand contract, instead.

Special note for custom interface implementations and mocking:

The default implementation of `ContentObjectSiteAspect.isIgnoreUpdates()` still exists to ease implementation and mocking in tests, but the default implementation changed to always return `false` instead of forwarding to `ContentObjectSiteAspect.getIgnoreUpdates()`.

[CMS-20066]

Third-Party Update: Apache Tika and Transitive Dependencies

Apache Tika has been updated to new major version 2.1.0.

As part of this change, the following transitive dependencies of Apache Tika have also been updated to match versions used by Tika:

- `com.adobe.xmp:xmpcore: 6.1.11`
- `com.drewnoakes:metadata-extractor: 2.16.0`
- `com.rometools:rome: 1.16.0`
- `commons-io:commons-io: 2.10.0`
- `org.apache.james:apache-mime4j-core: 0.8.4`
- `org.apache.james:apache-mime4j-dom: 0.8.4`
- `org.bouncycastle:bcmail-jdk15on: 1.69`
- `org.bouncycastle:bcprov-jdk15on: 1.69`
- `org.ow2.asm:asm: 9.2`

If you use these libraries in project code, please check their respective release notes for changes and upgrade information. No changes were necessary in the CoreMedia Blueprint for these updates.

Apache Tika restructured its Maven artifacts and split up the `tika-parsers` artifact from previous versions into separate artifacts. If you use Tika in project code, you may have to adapt Maven dependencies accordingly.

Apache Tika also removed some deprecated metadata keys [TIKA-1974]. If you have configured the *Content Feeder* or *CAE Feeder* to extract metadata values with configuration properties `feeder.tika.append-metadata` or `feeder.tika.copy-metadata`, then make sure that the configured metadata keys are still supported by Tika. You can find metadata keys supported by Tika in its API documentation, for example in interface `org.apache.tika.metadata.TikaCoreProperties`.

[CMS-20039]

Removal: Deprecated BaseCommandLineClient#addOptionValidated Method

The deprecated method `com.coremedia.cmdline.BaseCommandLineClient#addOptionValidated` has been removed from the Java API. If you still use it in your project code, switch to `options.addOption(option)`, `options` and `option` referring to the arguments of `addOptionValidated`.

[CMS-20025]

Removal: Deprecated Site#getManagerGroupName Method

The deprecated method `com.coremedia.cap.multisite.Site#getManagerGroupName` has been deleted. If you still use it in your project code, replace it with `getManagerGroupNames().stream().findFirst().orElse("")` -- mind, though, that it is better respecting all group names returned by `getManagerGroupNames()`.

[CMS-20003]

Removal: Deprecated Options of Some Commandline Tools

The command line tools `publish`, `destroy` and `query` do not support the option `paths` any longer. Use `path` instead (also CSV capable, despite the singular name). `paths` has already been undocumented for a while, so you might not even be aware that it existed.

[CMS-19998]

Removal: Deprecated AbstractVersionSelector#addOption Method

The deprecated method `com.coremedia.cmdline.management.AbstractVersionSelector#addOption` has been removed from the Java API.

If you still use it in your project code, switch to `options.addOption(option)`, `options` and `option` referring to the arguments of `addOption`.

[CMS-19996]

Removal: Deprecated `CommandLineParameters#addOptionValidated` Method

The deprecated method `com.coremedia.cmdline.CommandLineParameters#addOptionValidated` has been removed from the Java API. If you still use it in your project code, switch to `options.addOption(option)`, `options` and `option` referring to the first two arguments of `addOptionValidated`.

[CMS-19993]

Removal: Deprecated `AbstractUAPIClient#enableOutVerbose` Method

The deprecated method `com.coremedia.cmdline.AbstractUAPIClient#enableOutVerbose` has been deleted from the Java API. If you are still using it in your project code, switch to `com.coremedia.cmdline.BaseCommandLineClient#enableVerboseLogging`.

[CMS-19992]

Removal: Deprecated `MarkupUtil#isEmptyMarkup` Method

`com.coremedia.xml.MarkupUtil#isEmptyMarkup` has been deleted. Use `#isEmptyRichText` (for CoreMedia RichText only) or `#hasText` (for markup of arbitrary grammars) instead.

[CMS-19990]

New Feature: Workflow Final Actions and Archiving of Aborted Workflows

Workflows can now declare custom final actions, which are executed after a workflow process has completed successfully or was aborted. To this end, a workflow definition can contain `<FinalAction>` elements with custom classes that implement the interface `com.coremedia.cap.workflow.plugin.FinalAction`.

Workflow definitions for publication, translation, and synchronization workflows have been adapted to use the new predefined final action `ArchiveProcessFinalAction` instead of the old `ArchiveProcess` task action. With this change, aborted workflow processes are archived in the same way as completed processes. Users can see their archived processes as finished workflows in Studio.

You have to re-upload these changed predefined workflow definitions using the "cm upload" tool to make use of the new functionality. The standard workflows `studio-simple-publication.xml`, `studio-two-step-publication.xml`, and `/com/coremedia/translate/workflow/synchronization.xml` can be uploaded by passing their name to the `-n` option of the `cm upload` tool. The workflow definition `translation.xml` from Blueprint directory `apps/workflow-server/modules/cmd-tools/wfs-tools-application/src/main/app/properties/corem/workflows` can be uploaded by passing the file name to the `-f` option of the `cm upload` tool.

The old `ArchiveProcess` action from package `com.coremedia.workflow.common.actions` has been deprecated. It's recommended to replace usages in custom workflow definitions with the `ArchiveProcessFinalAction` in the same way as in predefined workflow definitions.

For more details, have a look at section "Final Actions" in the *Unified API Developer Manual* and section "Customize Workflow Definitions" in the *Workflow Manual*.

[CMS-19768]

MongoDB Update to version 5.0.2

MongoDB has been updated to version 5.0.2 for the Docker Deployment. Note that an upgrade is only possible from a 4.4.4 version. An upgrade from an earlier version needs to be done successively. Please refer to the official MongoDB documentation on how to upgrade: <https://docs.mongodb.com/v5.0/release-notes/5.0-upgrade-standalone/>

[CMS-19757]

MediaStore Changes

Database

The database table `BlobData` has been enhanced with a `uuid` column. The UUID of a blob is optionally assigned and consumed internally by the `contentserver`.

Note, that it is not possible to start a Content Server of a previous release on the database after the new Content Server has adapted the database tables. You should always make a backup of the Content Server database before upgrading.

[CMS-19748]

Issue Search Filter Enabled By Default and Solr Schema Changes

The Solr index schema for *Content Feeder* and *CAE Feeder*, and the configuration of the *Content Feeder* was changed to enable the search filter for validation issues by default, and to also support Solr nested documents in the *CAE Feeder*.

See also the updated section "Searching for Content | Content Issues" in the Search Manual.

Solr Configuration Changes

The Solr index fields `_root_` and `_nest_path_` were added in the index schema for both *Content Feeder* and *CAE Feeder* indexes in `content/conf/schema.xml` and `cae/conf/schema.xml`. These files are located in the Blueprint below `apps/solr/modules/search/solr-config/src/main/app/configsets`.

The fields were optional since version 2104 but not defined by default. Adding them requires recreation of indexes from scratch. If you create new indexes with this version, or if you have already created new indexes with previous versions and these fields defined, then you should add/keep the fields in the schema. For the *Content Feeder* index, these fields are required if indexing of validation issues is enabled. It's important that you don't change the definition of these fields, if you keep an existing index.

Content Feeder Configuration

The configuration of the *Content Feeder* was changed to index issues, by default. You can disable issue indexing with configuration property `feeder.content.issues.index=false`, for example if you have decided to not add the fields to the Solr schema to avoid reindexing, or if you are not using the default Solr integration but a custom `Indexer` that does not support indexing issues as nested documents.

[CMS-19606]

Solr Base Image added

A new base image has been added for the Solr search engine. The image can be found on DockerHub at [coremedia/solr-base](#). The image adds the scripting layer to setup a leader / follower setup in the standalone Solr mode.

The image also incorporates the change from the old master / slave terminology to the new leader / follower.

For the start the old environment variables will work as usually but they are deprecated and should be replaced with their new pendants.

The image also uses the default Solr image as its base as it is recommended by Solr.

The source repository for this image can be found at GitHub in our [coremedia-contributions](#) organisation in the [docker.solr-base](#) and access is granted on request to our support.

[CMS-19599]

Changed Solr Terminology: Leader/Follower replaces Master/Slave

Apache Solr has replaced the names of their replication setup from Master/Slave to Leader/Follower. These names have now been changed in Solr configuration files (`solrconfig.xml`) and CoreMedia documentation. For details of the change in Solr, see https://solr.apache.org/guide/8_8/solr-upgrade-notes.html#solr-8-7.

The following system properties used to configure Solr replication have been renamed. If you are using the old names in your project, you must update them accordingly:

- `solr.master.url` -> `solr.leader.url`
- `solr.master` -> `solr.leader`
- `solr.slave` -> `solr.follower`

Environment variables for the Docker image have also changed. The old environment variables still work, but are deprecated:

- `SOLR_MASTER` -> `SOLR_LEADER`
- `SOLR_MASTER_URL` -> `SOLR_LEADER_URL`
- `SOLR_SLAVE` -> `SOLR_FOLLOWER`

- SOLR_SLAVE_AUTOCREATE_CORES -> SOLR_FOLLOWER_AUTOCREATE_CORES
- SOLR_SLAVE_AUTOCREATE_CORES_LIST -> SOLR_FOLLOWER_AUTOCREATE_CORES_LIST
- SOLR_SLAVE_AUTOCREATE_THRESHOLD -> SOLR_FOLLOWER_AUTOCREATE_THRESHOLD

The Maven commands to start and stop a Solr follower node for local development in `apps/solr/modules/search/solr-config/pom.xml` have also changed to use "follower" instead of "slave". For details, see the comments in that `pom.xml` file.

In addition to the system properties and environment variables, the following changes have been applied:

- The default log-location for Solr followers changed: `slave-logs/` -> `follower-logs/`
- To start a Solr follower locally, you now need to call `mvn exec:exec@start-solr-follower` (instead of `mvn exec:exec@start-solr-slave`).

[CMS-19302]

Third-Party Update: Apache Solr 8.8.2

Apache Solr has been updated to version 8.8.2, which fixes some security vulnerabilities of the previous version.

Solr Configuration

Solr configuration files in `apps/solr/modules/search/solr-config/src/main/app/configsets` have been adapted for the new version. If you've customized these files, make sure to apply the change in `content/conf/solrconfig.xml` that sets `enableLazyFieldLoading` to false. This change is necessary to work around Solr bug SOLR-15192.

Solr Master/Slave Setup

For the Solr Master/Slave setup, the new Solr version now requires the configuration of hostname and port of the Solr master instance in the `solr.shardsWhitelist` system property of Solr slave instances (SOLR-15217). The Docker image build has been adapted accordingly in `apps/solr/docker/solr/src/docker/bin/config.sh`. See section "Master/Slave Index Replication" in the Search Manual or the Solr Reference Guide for more information.

For detailed changes of Apache Solr have a look at https://lucene.apache.org/solr/8_8_2/changes/Changes.html and https://solr.apache.org/guide/8_8/solr-upgrade-notes.html.

[CMS-19266]

Content Server and Workflow Server do not use `corem.home` anymore

The `corem.home` directory and system property are no longer used by the Content Server and Workflow Server.

For the Content Server a new Spring property `cap.server.base-dir` has been introduced to control the base directory for other configuration properties when using relative paths. By default, the property is set to the value of the `user.dir` system property. For the Blueprint Docker images this is the `/coremedia` directory. The following properties have been adapted and support relative file paths (some defaults have changed):

- `cap.server.license` (file / url, default: `license.zip`)
- `cap.server.document-types` (file / url patterns, default: `classpath*:framework/doctypes/**/*.xml`)
- `cap.server.login.authentication` (file / url, default: `classpath:coremedia-jaas.conf` resp. `classpath:blueprint-jaas.conf`)
- `cap.server.login.bouncers` (file, default: empty)
- `cap.server.encrypt-passwords-key-file` (file, default: `etc/keys/DATABASE_NAME.DATABASE_USER.rijndael`)
- `replicator.tmp-dir` (directory, default: `var/tmp`)

The Workflow Server has only one new property which has not been configurable before, that allows a relative file-path, which is now relative to the value of the `user.dir` system property:

- `workflow.working-dir` (directory, default: `var/tmp`)

The Workflow Server does not use `cap.client.server.ior.url` anymore, you have to use `workflow.ior-url` instead.

The command-line-tools still use `corem.home`, and as the Docker image for the Workflow Server also contains the workflow converter tool, it still sets `corem.home` but it is not used in the server itself.

The deprecated method `CapSystemInfo#getInstallationPath` has been removed (see CMS-19437).

The following modules that were used to create a `corem-home` directory structure for the servers have been removed:

- `content-server-config`

- `content-server-blueprint-config`
- `workflow-server-config`
- `workflowserver-blueprint-config`

[CMS-19112]

Commerce Validator Issue Feeding

Commerce related validator issues defined in `LcStudioValidatorsConfiguration` are now also feeded into the Solr index "studio" by default. Therefore the *Content Feeder* now also uses *Commerce Hub* connections to configured *Commerce Adapters*. The legacy [and deprecated] LiveContext Blueprint extension for IBM/HCL B2B is **not** supported.

General Problems reported as INFO

Additionally, the `ContentLinkValidator` and its subclasses do not report Commerce Connection issues ["StoreContextNotFound"] as *warning*, anymore, but as *info*. This was changed because otherwise lots of issues with level *warning* are persisted for each and every commerce related content item, although the problem is more generic and is not caused by the single content items. As a rule of thumb, issues that are fixable on the content itself are *warning* or *error* and more global issues, like commerce connection problems, are *info* level.

Performance Advise

Please be aware that for repositories with huge amounts of commerce-related content (for example, Augmented Pages, Augmented Categories, Augmented Products or Product Teasers), feeding might take much longer, especially if the commerce adapter cache is empty and remote calls to the commerce system need to be made.

You might want to consider to limit the duration of calls to the commerce adapter(s) by setting the application property `commerce.hub.data.deadline` to a lower value (default is 20s). If this deadline/timeout is reached, the validator will get an exception and reports an issue on level *INFO* with the code `<ContentType>_catalogError` [e.g. `CMProductTeaser_catalogError`].

[CMS-18965]

Removal: Deprecated Search and Feeder Configuration Properties

Several deprecated configuration properties have been removed for the *Content Feeder*, the *CAE Feeder* and for applications that connect to Solr for searching.

If you still use any of these properties, you have to replace them with the property that is listed right next to it in the following list (sorted alphabetically):

- `feeder.backgroundFeed.delay` -> `feeder.content.background-feed-delay`
- `feeder.executorQueueCapacity` -> `feeder.core.executor-queue-capacity`
- `feeder.executorRetryDelay` -> `feeder.core.executor-retry-delay`
- `feeder.indexDeleted` -> `feeder.content.index-deleted`
- `feeder.indexGroups` -> `feeder.content.index-groups`
- `feeder.indexNameInTextBody` -> `feeder.content.index-name-in-textbody`
- `feeder.indexReferrers` -> `feeder.content.index-referrers`
- `feeder.management.password` -> `feeder.content.management.password`
- `feeder.management.user` -> `feeder.content.management.user`
- `feeder.maxBatchBytes` -> `feeder.batch.max-bytes`
- `feeder.maxBatchByteSize` -> `feeder.batch.max-bytes`
- `feeder.maxBatchSize` -> `feeder.batch.max-size`
- `feeder.maxOpenBatches` -> `feeder.batch.max-open`
- `feeder.maxProcessedBatches` -> `feeder.batch.max-processed`
- `feeder.partialUpdate.aspects` -> `feeder.content.partial-update-aspects`
- `feeder.retryConnectToIndexDelay.seconds` -> `feeder.content.retry-connect-to-index-delay`
- `feeder.retrySendIdleDelay` -> `feeder.batch.retry-send-idle-delay` (*see note below*)
- `feeder.retrySendMaxDelay` -> `feeder.batch.retry-send-max-delay` (*see note below*)
- `feeder.sendIdleDelay` -> `feeder.batch.send-idle-delay` (*see note below*)
- `feeder.sendMaxDelay` -> `feeder.batch.send-max-delay` (*see note below*)
- `feeder.tika.timeout.milliseconds` -> `feeder.tika.timeout`
- `feeder.tika.warn.milliseconds` -> `feeder.tika.warn-time-threshold`
- `feeder.updateGroups.background.delay` -> `feeder.content.background-feed-delay`
- `feeder.updateGroups.immediately` -> `feeder.content.update-groups-immediately`

- `solr.collection.cae` -> `solr.cae.collection`
- `solr.collection.content` -> `solr.content.collection`
- `solr.configSet` -> `solr.cae.config-set` (for the *CAE Feeder*), `solr.content.config-set` (for the *Content Feeder*)
- `solr.partialUpdates` -> `feeder.solr.partial-updates.enabled`
- `solr.partialUpdatesSkipIndexCheck` -> `feeder.solr.partial-updates.skip-index-check`

Note, that the old properties `feeder.sendIdleDelay`, `feeder.sendMaxDelay`, `feeder.retrySendIdleDelay`, and `feeder.retrySendMaxDelay` took a value in seconds for the *Content Feeder*, but a value in milliseconds for the *CAE Feeder*. The new properties `feeder.batch.send-idle-delay`, `feeder.batch.send-max-delay`, `feeder.batch.retry-send-idle-delay`, and `feeder.batch.retry-send-max-delay` take milliseconds for both *Content Feeder* and *CAE Feeder* (if the value does not specify a different Spring Boot duration unit like 's').

[CMS-17959]

Improved CAE Feeder Database Usage

The *CAE Feeder* was improved to store data more efficiently in the database, which reduces database disk space requirements and improves throughput. To this end, the data type of some database columns was changed from a string type to a binary type. This change also applies to custom applications based on the *Proactive Engine*.

When the *CAE Feeder* is started with an empty database to feed everything from scratch, it will automatically create tables with improved definitions. If started with existing database tables from a previous release, then the *CAE Feeder* will work as before without improved database usage. Existing database tables are not upgraded. For best performance, it is recommended to start feeding with an empty database.

[CMS-16722]

Removal: Deprecated Watchdog and Probedog infrastructure

The Watchdog and Probedog components have been removed. To implement the same functionality you can use Spring Boot actuators together with an orchestration solution. In addition to the standard actuators of Spring Boot, CoreMedia adds a set of custom endpoints to represent the health state of the applications. You find the documentation in the operations basics handbook.

This update also removes the default access restrictions for the formerly inbuilt `watchdog` user. In case of an update it is recommended to remove this user manually to prevent security breaches.

[CMS-7318]

Improved Translation Workflow Auto-Merge

The "Auto-Merge" feature used in translation workflows to copy property values from master to derived content items has been improved. In previous releases, it completely ignored translatable properties and never changed such properties in the derived content. This has been changed because it could lead to unexpected results, especially for properties nested in structs. Furthermore, the merge algorithm was improved for link annotations.

Translatable (nested) property values are no longer excluded from auto-merge, but will be copied from the master to the derived content in the same way as other values. However, opposed to non-translatable properties, if the derived content already contains a different value for a translatable property, for example some translated text, then the value of the derived content will be kept and no warning about a merge conflict will be shown. Furthermore, for annotated links like teaser targets with "call-to-action" texts or channel page grid links with validity dates: if a link is removed or replaced in the master content, then the auto-merge feature will also remove or replace the link correctly in the derived content, even if some annotation properties of that link were changed in the derived content. With previous releases some annotation values could remain, even if the actual link had been removed. This has been fixed.

The previous behavior can still be restored by setting the following Spring configuration properties for the Workflow Server as follows:

- `workflow.localization.auto-merge.translatable=false`
- `workflow.localization.auto-merge.legacy-list-merge=true`

These properties are documented in the Deployment Manual, section "CoreMedia Properties Overview | Workflow Server Properties".

[CMS-19187]

Secure Blob Access

Blob access by clients is protected against URL guessing now. If you want to use components from earlier CMCC releases with new content servers, you must set the application property `cap.server.blob-md5-permission-check` to `false` for the new content servers.

[CMS-7021]

blueprint-doctypes-xmlrepo.xml: Available in Blueprint Sources

The `blueprint-doctypes-xmlrepo.xml` located in so-called Blueprint-Base, unavailable to most customers, now has a copy in the Blueprint module `test-util`.

All tests in Blueprint have been adapted to use this new location for the content-type definitions.

Old coordinates:

- Reference:

```
classpath:com/coremedia/blueprint/base/testing/blueprint-doctypes-xmlrepo.xml
```

- Dependency:

```
<dependency>
  <groupId>com.coremedia.blueprint.base</groupId>
  <artifactId>bpbse-test-util</artifactId>
  <scope>test</scope>
</dependency>
```

New coordinates:

- Reference:

```
classpath:com/coremedia/testing/blueprint-doctypes-xmlrepo.xml
```

- Dependency:

```
<dependency>
  <groupId>com.coremedia.blueprint</groupId>
  <artifactId>test-util</artifactId>
  <scope>test</scope>
</dependency>
```

Deprecation Notice:

The artifact `com.coremedia.blueprint.base:bpbase-test-util` is herewith deprecated for use in CoreMedia Blueprint. For the lifecycle of 2110 it no incompatible changes will be introduced to this deprecated artifact.

It is strongly recommended migrating your own tests to the new copy within `com.coremedia.blueprint:test-util`.

[CMS-2487]

5.5.2 CoreMedia Content Application Engine

Removal: CAE Module `cap-editing-pbe-taglib-component`

The unused and deprecated module `cap-editing-pbe-taglib-component` has been removed. It is recommended to use the tags `cm:previewScripts` and `cm:metadata` of the CAE tag library in JSP templates instead of the removed tags `pbe:head` and `pbe:contextinfo`, respectively.

[CMS-20438]

Moved static CAE resources to `'/static/'`

Build-in static CAE resources were moved to `'/static'`. Although this it is not a hard requirement that static resources be served from the `'/static'` path we do recommend to serve static resources using this prefix because it supports a clean separation of static resources from servlet mappings when developing rewrite rules.

[CMS-20436]

Deprecation of Image Operations

The image operation `gamma` and the parameters of the `unsharpMask` are now deprecated. Both operations couldn't be controlled by editors in the Studio directly for single images. They also haven't been used in the Blueprint by default.

While scaled images generally benefit from mild sharpening by means of `unsharpMask` and its default values, the parameters leak implementation details that are likely to change. `unsharpMask` as a non-parameterized part of the transformation string will remain available.

`gamma` will be removed in the future. If you are using this operation in your project, please let us know.

[CMS-20342]

Removal: CAE Code Minification

The deprecated and unused interface `com.coremedia.blueprint.cae.view.processing.Minifier` has been removed. It is recommended to minify and optimize `CSS` and `JavaScript` files during build steps in the frontend workspace. This will be done by the blueprint default webpack configuration. Therefore the property `disableServerSideCompression` has been removed from Theme Configuration and build scripts too.

[CMS-20323]

Removal: Elastic Social Personalization Extension

The Elastic Social Personalization extension was removed without replacement.

[CMS-20295]

Deprecation: Property "ieExpression" of `CMAbstractCode`

The doctype property `ieExpression` of `CMAbstractCode` has been deprecated due to removal of Internet Explorer support. All usages will be removed soon.

[CMS-20143]

Removal: Deprecated Sitemap URI Pattern `/service/sitemap/{sitemapPath}`

The deprecated Sitemap URI Pattern `/service/sitemap/{sitemapPath}` and the related handler have been removed from `com.coremedia.blueprint.cae.sitemap.SitemapHandler` [`cae-base-lib`]. The same applies to private members `ACTION_NAME` and `SITEMAP_PATH` which are now obsolete.

Use `{{/service-sitemap-\\{siteld\\}-\\{sitemapFile\\}}}` instead. It is defined by `SitemapHandler.URI_PATTERN_SITEMAP`.

If you still require the old mapping or members, ensure to keep your existing `SitemapHandler` in CoreMedia Blueprint.

[CMS-20048]

Removal: Deprecated UriComponentsHelper#addQueryParameters Method

The deprecated method `com.coremedia.objectserver.web.links.UriComponentsHelper#addQueryParameters` has been deleted. If you still use it in your project code, switch to `UriComponentsBuilder#queryParams (MultiValueMap)`

[CMS-20001]

Deprecation: ContentBlobHandlerBase

`com.coremedia.objectserver.web.ContentBlobHandlerBase` has been deprecated. No handlers use it any longer. If you still need it in your project code, please contact the CoreMedia support.

[CMS-19984]

Removal: Deprecated ContentBlobHandlerBase#URIVARIABLE_VERSION Field

The deprecated field `com.coremedia.objectserver.web.ContentBlobHandlerBase#URIVARIABLE_VERSION` has been deleted from the Java API. The result map of `ContentBlobHandlerBase#buildBlobLink` contains no entry for this key any longer. If you still use it in your project, migrate to `ContentBlobHandlerBase#URIVARIABLE_ETAG`.

[CMS-19983]

Removal: BlueprintFreemarkerFacade functions

Both deprecated Freemarker functions `getContainerMetadata()` and `getDisplaySize()` have been removed. Please use `Container#getContainerMetadata()` and `getDisplayFileSize()` instead.

[CMS-19958]

Removal: Deprecated Methods of CMProductList

The deprecated Methods from the `CMProductList` interface have been removed. The implementation of those methods are still available in '`CMProductListImpl.java`' as they are used in '`livecontext-dataviews.xml`'.

[CMS-19409]

Removal: CAE TokenResolver Class

Classes `com.coremedia.objectserver.web.links.TokenResolver` and `com.coremedia.objectserver.web.links.TokenResolverHelper` were removed because token resolving repeatedly caused encoding issues with links. Use Spring utilities to build URI components and their built-in support to replace variables instead.

[CMS-19163]

CAE cache-control default behavior changed

In CAE responses the "s-maxage" value in Cache-Control header was forced to be set to the same value as for "max-age", regardless of whether there was a separate configuration for "s-maxage". A factor can now be configured for setting the "s-maxage" header value in relation to "max-age" (see new property "`cae.cache-control.s-maxage-factor`"). Default value is "0.0" which means that no "s-maxage" is set unless it is explicitly configured. The old behavior can be restored by setting the value "1.0".

[CMS-19104]

Removal: LiveContext IBM/WCS implementation

The deprecated LiveContext HCL/IBM extension in the blueprint has been removed. It is replaced by the Commerce Hub. Connections to an eCommerce system are no longer established directly, but only via Commerce Hub/Adapter. The example Aurora B2B sites have been removed from the demo content. As a result, a few features of the B2B support have also been omitted. Note, there is still no replacement in the Commerce Hub for them. In detail, these are the handling of workspaces and contracts.

[CMS-19089]

Moved internal CAE controllers to management application context

In CM10 the CAE exposed several endpoints below ' `/blueprint/servlet/internal` ' which had to be handled separately when setting up URL rewrite rules so that they cannot be accessed from the public internet. In CM11 they are now configured to be included in the spring boot management context (together with the actuators) and are available on a different port (by default in blueprint deployment port 8081 on the CAE Docker Container and mapped to 40981 for preview and 42181 for live CAE). Affected endpoints are concerned with URL generation (`/blueprint/servlet/internal/service/url`), sitemap generation (`/blueprint/servlet/internal/<siteid>/sitemap-org`) and test URL generation (`/blueprint/servlet/internal/<siteid>/testurls`).

If use them in your project, please make sure to access them from your internal network using the new port and path without "`/blueprint/servlet`" prefix. For example: `http://<hostname>:42181/internal/service/url` for accessing the URL generation on the Live CAE

[CMS-18333]

Removal: BlueprintFreemarkerFacade functions

Both deprecated Freemarker functions `createBeanFor()` and `createBeansFor()` have been moved from `BlueprintFreemarkerFacade` to `LivecontextFreemarkerFacade` , because of the remaining usages in livecontext templates. We recommend to not use the functions in projects.

[CMS-15880]

Optimize Rendering of embedded content in Richtext

When embedding content (esp. images) in richtext, `asRichtextEmbed` templates are likely to break HTML validity by rendering `<div>` blocks inside the richtext's `<p>` context. So far, the Blueprint's `PDivUntanglingFilter` tried to be smart and adjust the HTML accordingly by closing and reopening the `<p>` context around an embedded object. However, this magic never really matched projects' expectations about the result, because it sometimes led to unwanted additional paragraphs or empty links.

Therefore, we removed the p/div adjusting features from the `PDivUntanglingFilter` and leave it to the `asRichtextEmbed` templates to provide HTML fragments that are suitable to be used within a `<p>` context. Consequently, we renamed the `PDivUntanglingFilter` to `EmbeddingFilter`.

If the old `PDivUntanglingFilter` is useful for you, you can keep it in your project code, and also preserve the `LinkEmbedFilter` and `ImageFilter` classes in their old versions.

The following templates in the example bricks and themes are changed:

- `CMTeasable.asRichtextEmbed.ftl` closes and opens the `<p>` elements before and after the include manually.
- `CMPicture.asRichtextEmbed.ftl` changed from `<div>` to ``.
- `CMPicture.media.ftl` uses the `<picture>` element for the responsive container instead of a `<div>`.

[CMS-19278]

Removal: Sitemaps generation for commerce sites

The sitemap generation for commerce sites is removed from the CAE. The implementation was part of the old IBM livecontext extension that was also removed. The feature can be re-implemented by integrating the `com.coremedia.livecontext.ecommerce.ibm.cae.sitemap` package from CMCC 10 blueprint.

[CMS-19247]

Removal:

CMHasContextsImpl#getDataViewFactory() method

CMHasContextsImpl#getDataViewFactory() has been removed, since its usage could lead to cyclic dependencies or uncontrollable cache invalidations.

[CMS-18512]

Richtext Pagination

ParagraphHelper#createParagraphs(Markup, int) was supposed to paginate Richtext at headings [<p> elements with class attributes p--heading-1-4] ever since. Due to a bug, this feature was broken. Now, it is fixed, so you might observe more pages. This concerns especially CMTeasableImpl#getTextAsParagraphs . If you don't like the fixed pagination, omit the change in StAXBBlockElementPaginator.java:81 when you update the Blueprint.

[CMS-13015]

5.5.3 CoreMedia Studio Client

Removal: Deprecated TabManager Classes/Interfaces

The following deprecated classes/interfaces have been removed from `editor-components`:

- TabManager
- TabManagerImpl

The corresponding `getTabManager` in `IEditorContext` and `EditorContextImpl` has also been removed.

[CMS-19921]

Removal: CommerceConnection#getVendorXXX Methods

The following methods of the interface `com.coremedia.livecontext.ecommerce.common.CommerceConnection` were removed without replacement:

- `getVendorProperties`
- `getVendorVersion`
- `getVendorUrl`

[CMS-19750]

Removal: Studio Packages Proxy App

The Studio Packages Proxy App has been removed. The new plugin mechanism can be used instead. See documentation for details.

[CMS-19698]

For contents having a "data"-property, the links for those blobs are now rendered instead of simple content links

Render links for blobs in the headless richtext instead of the previous content links which weren't optimal for content types like `CMDownload` or `CMPicture` where you might just be interested in the link to the data.

[CMS-19669]

Removal: Legacy Search Facets API from Studio

The support for the legacy commerce search facets had been removed from the Studio client and backend. The facet editor for product lists does only support the multi-facet selection from now on.

Old documents that still contain the legacy single-facet format must be migrated manually. When migrating from CMCC10 you can use the migrate button in CMCC10 2104+ to migrate your legacy values before migrating to CMCC11. If you still have legacy single

facet values stored, a validator message is shown which gives the user a hint which value to select again.

[CMS-19295]

Moved: Modules Moved to new Package

The following modules have been moved to the new npm package `@coremedia/studio-client.main.ckeditor4-components`

- LinkAction.ts
- LinkActionBase.ts
- PasteContentAction.ts
- PasteContentActionBase.ts
- AddCKEditorPluginsPlugin.ts
- AddCKEditorPluginsPluginBase.ts
- CoreMediaRichTextArea.ts
- CoreMediaRichTextAreaBase.ts
- CustomizeCKEditorPlugin.ts
- RemoveCKEditorPluginsPlugin.ts
- RemoveCKEditorPluginsPluginBase.ts
- CKEditor_properties.ts
- PropertiesAction.ts
- PropertiesActionBase.ts
- RichTextAction.ts
- RichTextActionBase.ts
- RichTextActionToggleButton.ts
- RichTextMenuCheckItem.ts
- RichTextMenuCheckItemBase.ts
- RichTextMenuItem.ts
- TableAction.ts
- TableActionBase.ts
- TeaserOverlayPropertyField.ts
- TeaserOverlayPropertyFieldBase.ts
- TeaserOverlayStyleSelectorBase.ts

- RichTextField.ts
- RichTextFieldBase.ts
- RichTextArea.ts

[CMS-18571]

Deprecation: Configuration `hideIssuesButton` in Control Room

Deprecated configuration `hideIssuesButton` of `com.coremedia.cms.editor.controlroom.workflow.translation.DefaultTranslationWorkflowDetailForm` as it has no effect anymore. The property will be removed in the next major release

[CMS-18021]

BooleanPropertyField Label Changes

We fixed a bug where BooleanPropertyFields had no tooltip. For this, we changed an internal plugin of the field. To this end, resource bundles need to be adapted to display the correct field labels.

Example:

`CMPicture_localSettings.disableCropping_true_text` needs to be: `CMPicture_localSettings.disableCropping_text`

[CMS-20399]

ILinkListWrapper Manipulation Methods Are Now Asynchronous By Signature

We needed to change the behaviour of `#setLinks()`, `#addLinks()`, `#addLinksAtIndex()`, `#moveLinks()`, `#moveLinksToIndex()` and `#removeLinksAtIndex()` for some important implementations of `ILinkListWrapper` to be asynchronous. As a consequence, we changed the signature of these methods to return a Promise. In custom implementations where no asynchronicity is needed, the code can simply be changed as in the following example.

Before:

```
override public function setLinks(links:Array):void {
```



```
    getVE().setValue(links);  
}
```

After:

```
override public function setLinks(links:Array):Promise {  
    return new Promise(function (resolve:Function):void {  
        getVE().setValue(links);  
        resolve(links);  
    });  
}
```

Code that uses the methods does not necessarily need to be changed. Just note that the result is not immediately present after method invocation. Also, if you want to catch errors that happen during the method execution, you can no longer use a try/catch block but you have to use `Promise#_catch()` instead.

[CMS-19108]

Fixed Inconsistent Usage of `ILinkListWrapper#acceptsLinks`

The function `ILinkListWrapper#acceptsLinks` had unclear documentation and therefore was neither used nor implemented consistently.

1. The most common use case was to only pass the links to be added to the link list like the first sentence of the documentation applied.
2. Some usages also prepended the existing links of the link list to the links to add before passing the array to the function. This has implied that all links are meant to be replaced.

CoreMedia adopted all implementations to the most common use case [see 1] and added an optional second parameter named `replaceLinks` that can be used if the other usage was meant.

Upgrade Notes

1. In case you make use of the method `acceptsLinks` please check if you need to set the parameter `replaceLinks`.
2. In case you have implemented an own implementation of `ILinkListWrapper` please adjust your implementation of the `acceptsLinks` method accordingly to consider the `replaceLinks` flag. It is also expected that a check against `getFreeCapacity` [or if `replaceLinks` is set to `true` against

`getTotalCapacity`] is made to make sure the links to be added fit into the link list.

[CMS-18849]

Simplified Most of the Configuration to [ExtConfig]

We simplified most of our configuration for Ext-Components from `Bindable` to `ExtConfig`. In case you are not working only with our Public API (and not, for example, with native JavaScript in Studio) this has no impact. Otherwise you might need to change access to configuration in your native JavaScript from "getSomeConfig" and "setSomeConfig" to "someConfig".

[CMS-16977]

5.5.4 CoreMedia Studio Server

Redesign of Jobs, Toasts and Notifications

- The running jobs window has been redesigned and aligned with the new Studio design.
- Former toast messages, that have popped up at button as green balloons are gone now. They have been replaced with a new toast API. These new toast appear at the lower left of the Studio and have a title, description and a severity.
- Notifications and the notifications menu have been redesigned and aligned with the new Studio design. Multiple notifications can be shown now and will be stacked. Additionally, they can be marked as unread now again.
- User settings have been added to hide toast and notifications.
- The redesign of the notification may require additional redesign for customized notifications.

[CMS-19563]

Added Search Filters for Library Extension

The Studio library does support search filters for extensions now. As a result, search filters can now be added for commerce integration. The existing search filters work as before and saved searches aren't affected by this change.

If custom search filters have been added to the library, the plugin rule declaration for adding this filter may have to change, depending on where the filter should be added. The library now has a separate search filter panel for every extension.

These are:

- *com.coremedia.cms.editor.sdk.collectionview.search.SearchFilters* for regular content
- *com.coremedia.catalog.studio.collectionview.search.CatalogSearchFilters* for the content based product catalog
- *com.coremedia.cms.editor.sdk.collectionview.search.CatalogSearchFilters* for commerce integrations
- *com.coremedia.cms.editor.sdk.collectionview.search.AssetSearchFilters* for the asset management

To add a custom filter, the usual *AddItemsPlugin* can be used on one of these panels. For example:

```
<editor:SearchFilters>
  <editor:plugins exml:mode="append">
    <ui:AddItemsPlugin>
      <ui:items>
        <ns:CustomFilter/>
      </ui:items>
    </ui:AddItemsPlugin>
  </editor:plugins>
</editor:SearchFilters>
```

Adding this code to a Studio plugin rules declaration will add the *CustomFilter* to the list of content filters. Also, the method *editorContext.getEnabledSearchFilterIds()* has been removed since simply adding a filter this way is sufficient now.

Removing existing filters works the same way: by using the *RemoveItemsPlugin* with the *itemId* of the corresponding filter.

[CMS-19094]

Validators

You can declare validators not only as Spring beans, but alternatively also by JSON configuration files now. This is supported for most core validator classes, except of those which make only sense as singletons and which are provided by default, like the `AvailableLocalesValidator`. Most Blueprint validator classes can easily be enabled for Json configuration in the project, if needed.

See Studio Developer manual / Customizing CoreMedia Studio / Server-Side Content Processing / Validators / Implementation and Activation of Validators for details.

Unfortunately, the option of JSON configuration required some API changes. The validator classes have been widely stripped off their Spring features, esp. `@Autowired` and `InitializingBean`. Not expecting validators to be Spring beans, but just simple POJOs, makes it much easier to instantiate them from JSON configuration files. And the Spring features that were actually used could easily be replaced by other means, like non-null constructor arguments in favor of `@Autowired`, setters and `afterPropertiesSet` checks.

The abstract base classes `PropertyValidatorBase` and `ContentTypeValidatorBase` have been replaced by the Spring-free `AbstractPropertyValidator` and `AbstractContentTypeValidator`, respectively. This affects the constructor signatures and setters of their subclasses, since mandatory configuration is required as constructor arguments now, in favor of setters. The changes are most obvious in the `ValidatorsConfiguration` class in the Blueprint.

If you want to use JSON configuration also for custom validators, the validator classes should refrain from using Spring features like `@Autowired` or `InitializingBean`. Otherwise, you must implement more complex validator factories.

[CMS-18931]

CatalogAlias configuration for single catalog setups

Configured catalogAlias below `commerce#catalogConfig#catalogAlias` is used to store commerce ids from now on.

Nothing needs to be changed,

- if you had nothing configured for `commerce#catalogConfig#catalogAlias` earlier or
- if `commerce#catalogConfig#catalogAlias` was configured as "catalog".

If you had configured something else, the alias was not used for your existing augmentations. Simply consider to remove the `catalogAlias` configuration then from `commerce#catalogConfig`.

[CMS-20303]

Use stable IDs in CAE Fragment Requests, if Possible

So far the `id` parameter of commerce fragment requests from the CoreMedia fragment connector to the CAE were treated as external technical IDs, whereas all other communication uses stable external IDs for commerce entities. This leads to unnecessary cache misses, commerce communication and memory consumption in the CAE and commerce adapter caches.

If possible the external ID should be used in fragment requests and also should be treated as external ID on the CAE side from now on. If the commerce system cannot pass the external ID into the fragment parameters, because only the technical ID is available, then this behavior must be configured on the commerce adapter side.

The property `"metadata.additional-metadata.allow-tech-ids=true"` has to be set for the commerce adapter, if you want to continue using the external technical ID for commerce entities in the fragment connector. For customer running a HCL setup the configuration `"metadata.additional-metadata.allow-tech-ids=true"` is set by default in the commerce adapter `wcs`, since the fragment parameters passed from the fragment connector contain technical entity IDs by default.

[CMS-19920]

Optimized Taxonomy Performance

The performance of the *DefaultTaxonomy* has been improved. This has been achieved through better UAPI caching. In addition, the property `maxDocumentsPerFolder` can be set now. This property will trigger the creation of new taxonomy documents created by the Taxonomy Manager in a new sub-folder named with a UUID once the default folder has reached the configured amount of documents. New sub-folders are created this way every time when the last used one has reached its limit.

Since the Spring configuration and some method signatures have changed for the class *DefaultTaxonomy.java*, custom integrations that extend from this class must be adapted.

[CMS-19418]

5.5.5 CoreMedia Workspace

Banned Jakarta EE 9 Artifacts

There is a mismatch between Maven coordinates and Java packages of Jakarta EE 8/9 artifacts. [For details see https://www.eclipse.org/community/eclipse_newsletter/2020/november/1.php.] We follow the strategy of Spring Boot 2.5 of how to deal with this problem and ban all Jakarta EE 9 artifacts. If you use Jakarta EE 9 artifacts in your project, please try to replace them with the according Jakarta EE 8 libraries.

[CMS-19484]

Removal: Deprecated Properties

Properties that were deprecated because they were replaced by Spring Boot configuration properties have now been removed:

- `aspect.cache.capacity` use `cache.capacities.com.coremedia.cae.aspect.Aspect` instead
- `assumesIdentity.cache.capacity` use `cache.capacities.com.coremedia.objectserver.dataviews.AssumesIdentity` instead
- `{{blueprint.sitemap.\}}*` use `{{cae.sitemap.\}}*` instead
- `caas.cache-capacities` use `cache.capacities.com.coremedia.caas.search.solr.SolrQueryCacheKey` instead
- `caas.context.trace.enabled` use `caas.stax.context-trace-enabled` instead
- `caas.persisted-queries.query-resources-exclude-pattern` use `caas.persisted-queries.exclude-file-name-pattern` instead
- `caas.persisted-queries.query-resources-map-pattern.apollo` use `caas.persisted-queries.apollo-query-map-resources-pattern` instead
- `caas.persisted-queries.query-resources-map-pattern.relay` use `caas.persisted-queries.relay-query-map-resources-pattern` instead

- `caas.querylist.search.cache.seconds` use `caas.querylist-search-cache-for-seconds` instead
- `cachekeysize.java.lang.object` use `cache.capacities.java.lang.Object` instead
- `cae.coderesources-max-age` use `{{cae.cache-control.\}}*` instead
- `cae.coderesources.maxAge` use `cae.coderesources-max-age` instead
- `cae.crossdomain.whitelist` use `{{cae.cors.allowed-origins-for-url-pattern\\{path:.\\}\\}}*` instead
- `cae.developer.mode` use `delivery.developer-mode` instead
- `cae.is.preview` use `delivery.preview-mode` instead
- `cae.is.standalone` use `delivery.standalone` instead
- `cae.preview.crossdomain-whitelist` use `{{cae.cors.allowed-origins-for-url-pattern\\{path:.\\}\\}}*` instead
- `cae.use.local.resources` use `delivery.local-resources` instead
- `cae.view.cycle-check` use `cycle-check` instead
- `cae.viewdispatcher.cache.size` use `cache.capacities.com.coremedia.objectserver.view.ViewLookup` instead
- `cap.procmon.enable` use `cap.server.enable-procmon` instead
- `cap.server.ORBServerHost` use `com.coremedia.corba.server.host` instead
- `cap.server.ORBServerPort` use `com.coremedia.corba.server.port` instead
- `cap.server.ORBServerSSLPort` use `com.coremedia.corba.server.ssl.ssl-port` instead
- `cap.server.blob.channel.timeout` use `cap.server.blob-channel-timeout` instead
- `cap.server.encryptpasswords.keyfile` use `cap.server.encrypt-passwords-key-file` instead
- `cap.server.groupcache.size` use `cap.server.cache.group-cache-size` instead
- `cap.server.groupcache.status.interval` use `cap.server.cache.group-cache-status-interval` instead
- `cap.server.http.port` use `cap.server.http-port` instead
- `cap.server.init.runlevel` use `cap.server.init-runlevel` instead

- `cap.server.memberfolderrightscache.size` use `cap.server.cache.member-folder-rights-cache-size` instead
- `cap.server.namingpolicy.allowat` use `cap.server.naming-policy-allow-at` instead
- `cap.server.repository.home` use `cap.server.repository-home` instead
- `cap.server.repository.system` use `cap.server.repository-system` instead
- `cap.server.resourcecache.size` use `cap.server.cache.resource-cache-size` instead
- `cap.server.resourcecache.status.interval` use `cap.server.cache.resource-cache-status-interval` instead
- `cap.server.rightscache.size` use `cap.server.cache.rights-cache-size` instead
- `cap.server.rightscache.status.interval` use `cap.server.cache.rights-cache-status-interval` instead
- `cap.server.session.ping.interval` use `cap.server.session-ping-interval` instead
- `cap.server.session.transient.threshold` use `cap.server.session-transient-threshold` instead
- `cap.server.termination.timeout.seconds` use `cap.server.termination-timeout-seconds` instead
- `cap.server.usercache.size` use `cap.server.cache.user-cache-size` instead
- `cap.server.usercache.status.interval` use `cap.server.cache.user-cache-status-interval` instead
- `cap.sysmon.enable` use `cap.server.enable-sysmon` instead
- `com.coremedia.corba.ORBRedirector.original.host` use `com.coremedia.corba.client.redirect.original-host` instead
- `com.coremedia.corba.ORBRedirector.original.port` use `com.coremedia.corba.client.redirect.original-port` instead
- `com.coremedia.corba.ORBRedirector.redirect.host` use `com.coremedia.corba.client.redirect.redirect-host` instead
- `com.coremedia.corba.ORBRedirector.redirect.port` use `com.coremedia.corba.client.redirect.redirect-port` instead
- `com.coremedia.corba.SSLClientSocketFactory.clearTextPort` use `com.coremedia.corba.client.ssl.clear-text-ports` instead

- `com.coremedia.corba.SSLClientSocketFactory.keystore` use `com.coremedia.corba.client.ssl.keystore` instead
- `com.coremedia.corba.SSLClientSocketFactory.passphrase` use `com.coremedia.corba.client.ssl.passphrase` instead
- `com.coremedia.corba.SSLClientSocketFactory.sslPort` use `com.coremedia.corba.client.ssl.ssl-ports` instead
- `com.coremedia.corba.SSLServerSocketFactory.keystore` use `com.coremedia.corba.server.ssl.keystore` instead
- `com.coremedia.corba.SSLServerSocketFactory.passphrase` use `com.coremedia.corba.server.ssl.passphrase` instead
- `com.coremedia.corba.SingleIpSocketFactory.ip` use `com.coremedia.corba.server.single-ip` instead
- `com.coremedia.orb.jndiName` use `repository.orb-jndi-name` instead
- `com.coremedia.transform.loadedImageCacheCapacity` use `cache.capacities.com.coremedia.transform.image.java2d.LoadedImageCacheKey` instead
- `contentBeanBackedPageGridPlacement.cache.capacity` use `cache.capacities.com.coremedia.blueprint.cae.layout.ContentBeanBackedPageGridPlacement` instead
- `counters.aggregationInterval.daily` use `counters.aggregation-interval-milliseconds.daily` instead
- `counters.aggregationInterval.daily` use `counters.aggregation-interval-milliseconds.daily` instead
- `counters.aggregationInterval.monthly` use `counters.aggregation-interval-milliseconds.monthly` instead
- `counters.aggregationInterval.monthly` use `counters.aggregation-interval-milliseconds.monthly` instead
- `counters.aggregationInterval.weekly` use `counters.aggregation-interval-milliseconds.weekly` instead
- `counters.aggregationInterval.weekly` use `counters.aggregation-interval-milliseconds.weekly` instead
- `counters.aggregationInterval.yearly` use `counters.aggregation-interval-milliseconds.yearly` instead
- `counters.aggregationInterval.yearly` use `counters.aggregation-interval-milliseconds.yearly` instead
- `counters.aggregationInterval` use `counters.aggregation-interval-milliseconds.for-all` instead

- `counters.aggregationInterval` use `counters.aggregation-interval-milliseconds.for-all` instead
- `db.assume.tables.exist` use `db.assume-tables-exist` instead
- `db.blob.tableoptions` use `db.blob-table-options` instead
- `db.constants.escapeBackslash` use `db.escape-backslash` instead
- `db.function.max.ignoresIndex` use `db.function.max-ignores-index` instead
- `db.function.modulo.infix` use `db.function.modulo-infix` instead
- `db.modules.textEngine` use `db.text-engine` instead
- `db.schema.memberName.length` use `db.schema.member-name-length` instead
- `db.schema.name.length` use `db.schema.name-length` instead
- `db.schema.resourceName.length` use `db.schema.resource-name-length` instead
- `db.schema.schema.name.uppercase` use `db.schema.name-uppercase` instead
- `db.select.fromSubselectNamed` use `db.from-subselect-named` instead
- `db.sgml.buffer.length` use `db.sgml-buffer-length` instead
- `db.type.bit.convert` use `db.type.bit-convert` instead
- `db.type.bit.false` use `db.type.bit-false` instead
- `db.type.bit.true` use `db.type.bit-true` instead
- `db.type.varchar.lengthInCharacters` use `db.type.varchar-length-in-characters` instead
- `db.update.supportsNamedTable` use `db.update-supports-named-table` instead
- `editorial.comments.data.cache.comments-cache-size` use `cache.capacities.com.coremedia.editorial.comments.data.editorial.comments` instead
- `editorial.comments.data.cache.qualifiers-cache-size` use `cache.capacities.com.coremedia.editorial.comments.data.qualifiers` instead
- `editorial.comments.data.cache.threads-cache-size` use `cache.capacities.com.coremedia.editorial.comments.data.commentThreads` instead

- `filter.viewlookup.by.predicate` use `cae.view.filter-lookup-by-predicate` instead
- `livecontext.crossdomain.whitelist` use `{{cae.cors.allowed-origins-for-url-pattern\\(\\{path:.\\}\\}}*` instead
- `metadata.enabled` use `cae.preview.metadata-enabled` instead
- `models.createIndexes` use `mongodb.models.create-indexes` instead
- `mongoDb.clientURI` use `mongodb.client-uri` instead
- `mongoDb.host` use `mongodb.host` instead
- `mongoDb.prefix` use `mongodb.prefix` instead
- `page.cache.capacity` use `cache.capacities.com.coremedia.blueprint.common.contentbeans.Page` instead
- `pageGridImpl.cache.capacity` use `cache.capacities.com.coremedia.blueprint.cae.layout.PageGridImpl` instead
- `pbe.studioUrlWhitelist` use `cae.preview.pbe.studio-url-whitelist` instead
- `publisher.connection.timeoutSeconds` use `publisher.connection-timeout-seconds` instead
- `publisher.linkValidator.class` use `publisher.link-validator-class` instead
- `publisher.status.interval` use `publisher.status-interval` instead
- `replicator.Packager.flushSize` use `replicator.packager-flush-size` instead
- `repository.connect.retry.delay.seconds` use `repository.connect-retry-delay-seconds` instead
- `searchTermExternalReferenceResolver.cacheCapacity` use `cache.capacities.com.coremedia.livecontext.fragment.resolver.SearchTermExternalReferenceResolver` instead
- `searchTermExternalReferenceResolver.cacheTime` use `cache.timeout-seconds.com.coremedia.livecontext.fragment.resolver.SearchTermExternalReferenceResolver` instead
- `sql.modules.textEngine` use `sql.modules-text-engine` instead
- `sql.store.collector.blob.preservationperiod` use `sql.store.collector.blob-preservation-period` instead
- `sql.store.convert.correct-preferences` use `sql.store.convert-correct-preferences` instead

- `sql.store.convert.correctRichText` use `sql.store.convert-correct-rich-text` instead
- `sql.store.convert.defaultHoxImageProperty` use `sql.store.convert-default-hox-image-property` instead
- `sql.store.convert.filterXmlChars` use `sql.store.convert-filter-xml-chars` instead
- `sql.store.sgmlcache.interval` use `sql.store.sgml-cache-interval-seconds` instead
- `sql.store.sgmlcache.sizeBytes` use `sql.store.sgml-cache-size-bytes` instead
- `sql.store.sgmlcache.size` use `sql.store.sgml-cache-size-bytes` instead
- `studio.security.autoLogout.delay` use `studio.auto-logout.delay` instead
- `taskQueues.executionTimeout` use `taskqueues.execution-timeout-milli-seconds` instead
- `taskQueues.executionWarningTimeout` use `taskqueues.execution-warning-timeout-milli-seconds` instead
- `taskQueues.numberOfRetries` use `taskqueues.number-of-retries` instead
- `taskQueues.pollingInterval` use `taskqueues.polling-interval-milli-seconds` instead
- `taskQueues.recoveryInterval` use `taskqueues.recovery-interval-milli-seconds` instead
- `taskQueues.retryInterval` use `taskqueues.retry-interval-milli-seconds` instead
- `taskQueues.workerNode` use `taskqueues.worker-node` instead
- `view.debug.enabled` use `cae.view.debug-enabled` instead
- `view.errorhandler.enabled` use `cae.view.errorhandler.enabled` instead
- `view.errorhandler.output` use `cae.view.errorhandler.output` instead
- `view.maxdepth` use `cae.view.max-depth` instead
- `viewdispatcher.cache.enabled` use `cae.viewdispatcher.cache.enabled` instead
- `viewdispatcher.cache.size` use `cae.viewdispatcher.cache.size` instead

- `viewdispatcher.expose-spring-macro-helpers` use `cae.viewdispatcher.expose-spring-macro-helpers` instead
- `viewdispatcher.fallback-to-default-view` use `cae.viewdispatcher.fallback-to-default-view` instead

[CMS-19116]

Updated to Spring 5.3.12 and Spring Boot 2.5.6

In order to benefit from the improvements of the latest version of the Spring framework and to prepare for making use of the new features various 3rd party libraries have been updated.

- Spring Boot 2.5.6
- Spring Data 2021.0.6
- Spring Framework 5.3.12
- Spring Security 5.5.3
- AssertJ 3.19.0
- AspectJ 1.9.7
- Caffeine 2.9.2
- Classmate 1.5.1
- Commons dbcp2 2.8.0
- Esapi 2.2.3.1
- Gson 2.8.8
- Hibernate 5.4.32.Final
- Hibernate Validators 6.2.0.Final
- Jackson 2.12.5
- Janino 3.1.6
- Logback 1.2.6
- Micrometer 1.7.5
- MySQL 8.0.27
- Netty 4.1.69.Final
- Postgres 42.2.24
- Slf4j 1.7.32
- Tomcat 9.0.54

- XMLUnit 2.8.3

For more details on the changes in the Spring Boot 2.5 and 2.4 releases, see <https://github.com/spring-projects/spring-boot/wiki/Spring-Boot-2.5-Release-Notes> and <https://github.com/spring-projects/spring-boot/wiki/Spring-Boot-2.4-Release-Notes>. Also see the Spring 5.3 GA announcement <https://spring.io/blog/2020/10/27/spring-framework-5-3-goes-ga> and the Spring Security 5.5 GA announcement <https://spring.io/blog/2021/05/18/spring-security-5-5-goes-ga>

Spring Web MVC now treats `null` arguments as missing, see <https://github.com/spring-projects/spring-framework/wiki/Upgrading-to-Spring-Framework-5.x#web-applications> and adapt your handlers accordingly.

Due to changes in the esapi library some unused methods of its `SecurityConfiguration` class had to be removed. You will most likely not recognize this change because the usage of this methods was not supported.

[CMS-18517]

Removal: repository.workflow, repository.cplist, and usecplist Properties

Removed the properties `repository.workflow`, `repository.cplist`, and `usecplist`. Use `repository.workflow.connect` instead of `repository.workflow` and `repository.cplist.connect` instead of `repository.cplist` and `usecplist`.

[CMS-17426]

Switched to Sencha CMD 7.2.0.84

Please use Sencha CMD 7.2.0.84 to build the studio-client to avoid heap space related problems after switching to the new NPM/TypeScript based tooling.

[CMS-20076]

Mongo shell replacement mongosh

The mongo shell has been deprecated with MongoDB 5.0. The replacement shell is mongosh. Usages of `mongo` shell have been replaced with `mongosh` in documentation, deployment scripts and the example script `dump-es-user-data.js`.

[CMS-19884]

Renamed groupId com.coremedia.content-hub to com.coremedia.cms

The groupId of the content-hub core artifacts has been changed from `com.coremedia.content-hub` to `com.coremedia.cms` in order to adjust to naming conventions.

[CMS-16969]

5.5.6 CoreMedia Frontend Workspace

Removal: IE11 Support in Themes and Bricks

The example themes and bricks in the frontend workspace do not support Internet Explorer 11 anymore by removing it from the `browserlist` entry in the `package.json`. Also extra polyfills for IE11 in the blueprint bricks and in the tooling have been removed. You can still add polyfills or custom styles for IE11 to themes and bricks in your projects. You may need to adapt the webpack and babel configuration.

[CMS-20138]

Changed Package Manager in Frontend Workspace from yarn to pnpm

The package manager of the frontend workspace has been changed from `yarn` to `pnpm`. `Yarn` in version 1 is not maintained anymore and `pnpm` is fast, efficient, strict and supports monorepos without additional libraries like `lerna`. Please check <https://pnpm.io/> for more details and installation. Please use the latest 6.x version of `pnpm`. Changes to the tooling should not affect any bricks or themes in your project.

Changes:

- Support for building the frontend workspace with `maven` or other package manager has been removed.
- The lock file for pinned versions changed from `yarn.lock` to `pnpm-lock.yaml`
- A new file for the workspace structure has been introduced: `pnpm-workspace.yaml`

- All workspace commands work like before, but just with `pnpm`. Examples:
 - `yarn install => pnpm install`
 - `yarn start => pnpm start`
 - `yarn build => pnpm build`

Upgrade steps: Since `pnpm` uses a non-flat `node_modules` structure, it is important to check for missing dependencies in custom themes and bricks. Please remove all `node_modules` in your project workspace. If a build fails, add the missing dependency to your theme or brick according to the error message.

Optionally: `Pnpm` uses a different versioning for local dependencies (it adds a prefix `workspace:` before the version). You can run `pnpm up -r` to change the entries in the `package.json` files. All local dependencies in the blueprint themes and bricks have been changed to an even more generic way, because versions inside the workspace are not important. Example: `"@coremedia/theme-utils": "^3.0.0" => "@coremedia/theme-utils": "workspace:*"`.

[CMS-19955]

Third-Party Update: Node.js 16

Node.js has been updated to the latest LTS version 16. Please update your local installation to version 16 for local development too. Node.js version 14 is not supported!

[CMS-19245]

5.5.7 CoreMedia Deployment

Update to Latest Commerce Adapter Versions and Adaption of Configurations

The [default] versions of the commerce adapters have been updated to the latest 2.0.x versions. The new adapter versions require some configuration changes:

Commerce Adapter Salesforce Commerce Cloud :

- The replacement tokens `sfra.host` and `sitegenesis.host` are not configured in the adapters default configuration anymore, because they are specific to the demo data that is shipped with the Blueprint. As a result, the `METADATA_REPLACEMENTTOKENS_SFRA_HOST` and `METADATA_REPLACEMENT`

`TOKENS_SITEGENESIS_HOST` must now be configured in the `commerce-adapter-sfcc.yml` in the Blueprint. The `SFCC_LINK_STOREFRONT_HOST` is not needed by the adapter anymore and has been removed.

Commerce Adapter SAP Commerce Cloud :

- The replacement token `hybris.host` is not configured in the adapters default configuration anymore, because it is specific to the demo data that is shipped with the Blueprint. As a result, the `METADATA_REPLACEMENTTOKENS_HYBRIS_HOST` must now be configured in the `commerce-adapter-hybris.yml` in the Blueprint. The `HYBRIS_LINK_STOREFRONTHOST` is not needed by the adapter anymore and has been removed.
- The default values for the OAuth credentials have been from the adapter configuration, hence the `HYBRIS_OAUTH_CLIENTID` and `HYBRIS_OAUTH_CLIENT_SECRET` must now be explicitly set in the deployment configuration.

Commerce Adapter HCL Commerce

- The replacement token `aurora.host` is not configured in the adapters default configuration anymore, because it is specific to the demo data that is shipped with the Blueprint. As a result, the `METADATA_REPLACEMENTTOKENS_AURORA_HOST` must now be configured in the `commerce-adapter-wcs.yml` in the Blueprint. The `WCS_LINK_STOREFRONT_HOST` is not needed by the adapter anymore and has been removed.

[CMS-20063]

Enable Readiness and Liveness Health Group Endpoints

With Spring Boot 2.3 it is possible to define two extra subpaths of the health actuator to be used as the probing endpoints of the liveness and readiness checks in Kubernetes. This change enables those groups and adds the relevant checks to these groups.

For the cae and the headless-server, this change adds the `uapiConnectionReadiness` health check to the readiness health group. The health check is responsible to switch its state, whenever the UAPI connection is and **will** become unavailable.

This allows for an automation to remove the clients from its ingress loadbalancing before its repository becomes unavailable but allows him to process all outstanding request during a grace period given to the repository on a runlevel switch to maintenance.

To disable this new behaviour, remove the following two properties from the `application.properties` in the cae-live-app, cae-preview-app and the headless-server-app:

- `management.health.uapiConnectionReadiness.enabled`
- `management.endpoint.health.group.readiness.include`

[CMS-19923]

Enables deployment of SFMC via docker-compose.

Includes `global/deployment/docker/compose/sfmc.yml` to deploy SFMC via docker-compose.

[CMS-19249]

Removal: Deployment Archive with Chef Deployment

The Deployment Archive has been removed. Please use the container images build by the `*-app` Maven modules to deploy CoreMedia. The Docker compose development setup, shows what images need to be used, what environment variables are mandatory and how ports and volumes need to be mapped.

[CMS-19157]

Container Image Build Process Changes

The `dockerfile-maven-plugin` has been replaced with the `google-jib-plugin` to build OCI conform container images, because the `dockerfile-maven-plugin` is no longer maintained. The `google-jib-plugin` integrates well with Spring-Boot applications and allows to build images securely without Docker being installed. This allows for a rootless, daemonless build but restricts the image building process to use only adding image layers by adding resources. There is no possibility to execute any `RUN` steps during the build process. Building the images without docker now requires a configured container registry although it is still possible to build images in the local docker daemon, without the need of a registry.

The switch to the new image build process is tightly coupled to the recent change to build all artifacts in a reproducible manner, resulting in identical image digests when no the source has not changed. With Spring Boot layered applications, this results in a faster build and smaller upload size to your container registry. Because file and image creation date affect the resulting image digest, jib sets the date to Unix epoch [00:00:00, January 1st, 1970 in UTC]. If you do not like this, you can enable the current timestamp

in the `jib-maven-plugin`, but this sacrifices reproducibility since the timestamp will change with every build.

```
<container>
  <useCurrentTimestamp>true</useCurrentTimestamp>
</container>
```

As a result of this change, the image build process moved to the application modules and the `docker` directory hierarchy has been removed.

All Spring-Boot application images are still based on the `coremedia/java-application-base` images will run with only some small breaking changes to the previous images. The breaking changes include:

- the application root directory `/coremedia` is now write protected as it is intended by the `google-jib-plugin`.
- The log files are now written to `/coremedia/log/application.log` if the `dev` profile is activated.

To build the images, the properties and Maven profiles have changed:

- the Maven profile to build the image is now named `default-image` with the property `jib.goal` to be either set to `build` or `dockerBuild`, depending on whether you want to use the Docker daemon or the registry as a build target.
- the property to define the registry has been renamed from `docker.repository.prefix` to `application.image-prefix`
- the property to define the image tag has been renamed from `docker.image.tag` to `application.image-tag`
- the property to define the base image has been renamed from `docker.java-application-base-image.repo` to `application.image-base`
- the property to define the image name has been renamed from `docker.repository.suffix` to `application.image-suffix`

The new images do not contain any health checks directive any more, if you rely on the docker health checks, please define them either on the command line or in your `docker-compose` file.

```
services:
  foo:
    test: [ "CMD", "curl", "-Lf", "http://localhost:8081/actuator/health" ]
    interval: 30s
    timeout: 10s
    retries: 3
```

```
start_period: 40s
```

```
docker run --health-cmd='curl -Lf http://localhost:8081/actuator/health ||  
exit 1' \  
  --health-interval=30s \  
  --health-timeout=10s \  
  --health-start-period=40s \  
  --health-retries=3 <image>
```

[CMS-18332]

Traefik update in Docker Setup

Traefik has been updated to 2.5.

With this update, we changed the configuration provider from Docker to File. This change has the benefit that:

- the Docker compose files become more readable and less complex.
- Traefik can be installed on a separate machine and only the service endpoints need to be reconfigured to point to the nodes, where the services are running.

The CAE rules have been changed from a whitelist pattern to only allow a specific set of servlets to a match-all style. This is possible, since the `internal` servlet has been moved to the server running on the maintenance port.

[CMS-16416]

5.5.8 CoreMedia Commerce Hub

Removal: Special Commerce Hub API for Preview URLs

The special treatment of Studio preview urls has been removed from Commerce Hub API. Now preview urls are generated the same way as the other commerce related links via LinkTemplates and dynamic replacement values. Most of the changes took place within the commerce adapter workspaces. On the CMS side the PreviewUrlService has been deleted and there are changes in the CommerceStudioLinks.

If you didn't do any customization regarding Studio Preview Urls for commerce items, nothing needs to be changed on your side.

[CMS-20308]

Updated Multi Catalog Configuration

The catalog alias configuration for the multi catalog feature has been updated. The former Studio content configuration setting "livecontext.catalogAliases" has been removed and needs to be updated, if you use multiple catalogs for one site. If you do not use the multi catalog feature there is no action required.

If you use multiple catalogs for one site, the catalog aliases are configured via the #additionalCatalogConfigs struct list below the commerce struct instead.

Old format:

```
<StructProperty Name="livecontext.catalogAliases">
  <Struct>
    <StringProperty Name="master">Extended Sites Catalog Asset
Store</StringProperty>
  </Struct>
</StructProperty>
```

New format:

```
<StructProperty Name="commerce">
...
  <StructListProperty Name="additionalCatalogConfigs">
    <Struct>
      <StringProperty Name="alias">master</StringProperty>
      <StringProperty Name="name">Extended Sites Catalog Asset
Store</StringProperty>
      <StringProperty Name="id">12345</StringProperty>
    </Struct>
  </StructListProperty>
...
</Struct>
```

Please consult the migration guide or the commerce adapter documentation for more details.

[CMS-20130]

Removal: Deprecated Methods and Implementation

Removed deprecated methods `getCapacity/getCacheTimesInSeconds` of class `com.coremedia.blueprint.base.livecontext.ecommerce.common.CommerceCache`. Also removed its `isEnabled` method because the commerce cache can no longer be disabled.

Removed the legacy augmentation implementation and property `livecontext.augmentation.backward-compatibility`. Augmentation now requires observed content properties.

[CMS-20114]

Removal: LC3 SearchService

Removed the interface `com.coremedia.livecontext.ecommerce.search.SearchService` because commerce hub does not support search suggestions. Search is supported via `com.coremedia.livecontext.ecommerce.catalog.CatalogService#search`.

[CMS-19704]

Removal: Deprecated Developer Feature Candy Shop URLs

The deprecated developer feature "Candy Shop URLs" has been removed from the Blueprint. There is no replacement planned, developers will need to configure their local machine in the shop configuration to retrieve fragments from their localhost. Frontend development is easily possible through the Frontend Developer workflow and uploading of local Themes to the CMS.

[CMS-19695]

CurrentStoreContext and UserStoreContext API change

Deprecated methods have been removed from `com.coremedia.blueprint.base.livecontext.ecommerce.common.CurrentStoreCon`

`text` and `com.coremedia.blueprint.base.livecontext.ecommerce.common.CurrentUserContext`.

`get`, `set` and `find` APIs need explicitly the `ServletRequest` as a parameter.

`HttpContext` and `StoreContext` dependent implementations have been modified with the following breaking changes

- `com.coremedia.livecontext.ecommerce.order.CartService` methods accept a `HttpContext` as an argument.
- `com.coremedia.livecontext.ecommerce.pl3n.SegmentService#findSegmentsForCurrentUser` accept a `HttpContext` as an argument.
- The methods `withHttpContext` and `getHttpContext` have been removed from `com.coremedia.blueprint.base.livecontext.client.common.GenericStoreContextBuilder`.
- Deprecated `getCatalogPicture`, `findPictures` and `findVisuals` methods have been removed from `com.coremedia.livecontext.ecommerce.asset.AssetService`

[CMS-19451]

Removal: Obsolete Base Classes from LiveContext Test Modules

With the introduction of the Commerce Hub, the connections to commerce systems have been moved away from the central CMS services to the commerce adapters. After all legacy `LiveContext` extensions (those, that don't use the commerce adapters) have been removed, their base test classes are now obsolete and have also been removed.

[CMS-19345]

Removal: Hoverfly Extension Classes from LiveContext Test Modules

With the introduction of the Commerce Hub, the connections to commerce systems have been moved away from the central CMS services to the commerce adapters. The tests that require commerce connections (whose requests and responses can be recorded and replayed with Hoverfly) have been moved, too, making the Hoverfly extension classes in `LiveContext` test modules obsolete, hence they have now been removed.

[CMS-19343]

Removal: Deprecated Legacy Search API from LiveContext Extensions

CMCC11 doesn't support old adapter versions with legacy search API anymore. Legacy commerce client API and implementations have been removed. This includes:

- support for product search API V1 [replaced by API V2 since CMS 2104.1]
- map-based search params [replaced by `com.coremedia.livecontext.ecommerce.search.SearchQuery` since CMS 2104.1]

[CMS-19290]

Removal: Deprecations from LiveContext API

The old LiveContext API is used to access the Commerce Hub / generic commerce client from Studio and other clients. Some deprecations have now been deleted that are not needed anymore or that are not supported by the Commerce Hub. They are in detail: the AvailabilityService [not supported by the CommerceHub, should be requested directly from the commerce system], UserServer and UserSessionService [not needed anymore, the whole part of session sync between the CMS and the commerce system is gone, only the login/logout can be done via the commerce adapter], the AssetUrlProvider [is also not needed anymore].

[CMS-19221]

Removal: Deprecated lc-extension Related LinkSchemes

Obsolete deprecated linkschemes used for commerce integration via lc-extension have been deleted.

[CMS-19215]

Removal:

`com.coremedia.blueprint.base.livecontext.service` Package

The deprecated `com.coremedia.blueprint.base.livecontext.service` package has been removed without replacement as it was part of the legacy Blueprint commerce integration. The Commerce Hub integration does not yet offer corresponding classes dealing with connections to Storefront services.

Class `com.coremedia.blueprint.base.livecontext.ecommerce.common.HttpClientFactory` has also been removed. Use one of the various 3rd party http clients instead.

[CMS-19214]

Removal: Deprecated Livecontext Classes and Methods

The classes `AbstractCommerceBean` and `SpringCommerceBeanFactory` from package `com.coremedia.blueprint.base.livecontext.ecommerce.common` were removed. Use `com.coremedia.blueprint.base.livecontext.client.beans.ClientCommerceBeanFactoryMethodRegistry` instead.

The method `com.coremedia.livecontext.ecommerce.common.CommerceConnection#getStoreContext` was removed, use `com.coremedia.livecontext.ecommerce.common.CommerceConnection#getInitialStoreContext` instead.

The class `com.coremedia.blueprint.base.livecontext.ecommerce.common.CommerceConnectionInitializer` was removed, use `com.coremedia.blueprint.base.livecontext.ecommerce.common.CommerceConnectionSupplier` instead.

The class `com.coremedia.blueprint.base.livecontext.ecommerce.common.StoreContextHelper` was removed, use `com.coremedia.blueprint.base.livecontext.ecommerce.common.CurrentStoreContext#get(javax.servlet.ServletRequest)` and its other convenience methods instead.

[CMS-19213]

Removal: Deprecated Commerce Cache Invalidation Implementation

The deprecated source code for the Commerce Cache Invalidation of the LiveContext 3 extensions has been removed. Customizations based on that implementation have to be updated to work with the commerce cache invalidation implementation of the commerce hub [in commerce adapters and in the commerce hub client implementation].

[CMS-19212]

Removal: The ForVendor-Methods from the CommerceConnection

Removed methods `com.coremedia.livecontext.ecommerce.common.CommerceConnection#getServiceForVendor` and `com.coremedia.livecontext.ecommerce.common.CommerceConnection#getQualifiedServiceForVendor`. Use class `com.coremedia.blueprint.base.livecontext.ecommerce.common.CommerceServicesForVendor` instead.

[CMS-19124]

Simplified Registration of Custom Commerce Bean Types

The class `com.coremedia.livecontext.ecommerce.common.CommerceBeanTypeRegistry` was removed and `com.coremedia.livecontext.ecommerce.common.CommerceBeanType` was converted from an interface to an implementation class. Instead of registering enum classes defining custom commerce bean types you may now define custom commerce bean types on demand by calling `com.coremedia.livecontext.ecommerce.common.CommerceBeanType#of (String)`. This solves the issue of class loading order which occasionally lead to missing custom commerce bean types at runtime.

The class `com.coremedia.blueprint.base.livecontext.client.beans.ClientCommerceBeanFactoryMethodRegistry` was removed from public API. Use `com.coremedia.blueprint.base.livecontext.client.beans.ClientCommerceBeanFactory#register` instead to register a custom commerce bean class.

[CMS-17159]

Removal: CommerceConnection#getClone Method

Removed `com.coremedia.livecontext.ecommerce.common.CommerceConnection#getClone`. Commerce Connection instances must not be modified after creation.

[CMS-15229]

5.5.9 CoreMedia Headless Server

GraphQL Java Libraries Updated to Version 17.3

These libraries have been updated:

- `com.graphql-java:graphql-java` from 16.2 to 17.3
- `com.graphql-java:graphql-java-extended-scalars` from 15.0.0 to 17.0
- `com.graphql-java:java-dataloader` from 2.1.1 to 3.1.1

The public API CoreMedia class `CacheMapWrapper`, used for the dataloader cache was revoked due to breaking changes in the dataloader API. The dataloader cache is now created using the updated dataloader API only. (see CaasConfig)

Since version 17.0, GraphQL Java checks the syntax of the schema more strictly. Types implementing from 2 or more interfaces, now have to use an "&" character (instead of a comma) to enumerate interfaces. If the schema was customized, the schema should be checked for potential breaking changes in the used syntax.

For details see the original release notes:

- <https://github.com/graphql-java/graphql-java/releases>
- <https://github.com/graphql-java/java-dataloader/releases>
- <https://github.com/graphql-java/graphql-java-extended-scalars/releases>

[CMS-20151]

Swagger Updated to Latest Version

Swagger and the related libraries were updated for maintainability reasons. The updates made some refactorings necessary. As a consequence, the previously public api package `com.coremedia.caas.web.rest` has been revoked, as it doesn't contain any classes anymore. The previous public 3rd party package `springfox` has been revoked also, as the CoreMedia classes do not expose the Springfox annotations anymore.

As the swagger default page address has changed from `/swagger-ui.html` to `/swagger-ui/index.html`, we've marked this change breaking.

The new libraries for swagger/open api are:

- `io.swagger.core.v3:swagger-annotations:jar:2.1.10`
- `io.swagger:swagger-annotations:jar:1.6.2`

The new library versions for the swagger ui are:

- `io.springfox:springfox-boot-starter:jar:3.0.0`
- `io.springfox:springfox-core:jar:3.0.0`
- `io.springfox:springfox-spi:jar:3.0.0`
- `io.springfox:springfox-spring-web:jar:3.0.0`

[CMS-19836]

Refactoring of Adapters

Due to an internal refactoring, some of the adapters used in the content-schema had to be refactored as well. Where possible we tried to be non breaking. Nonetheless some changes lead to breaking changes in the content-schema:

- `byPathAdapter`
- `pageGridAdapter`
- commerce extension: `productPageGridAdapter`
- commerce extension: `categoryPageGridAdapter`

Also affected by the refactoring is the class `SearchQueryHelper`: These methods were dropped:

- `public static String validFromPastToValueQuery(String fieldName)`
- `public static String validFromValueToFutureQuery(String fieldName)`

Please use the full signature methods with the additional call parameter `ZonedDateTime actualTime`.

This public API class was removed without a replacement, as the refactoring made it redundant: `ExecutionResultHandlerHelper`

[CMS-19796]

Filter Predicates May Now Access the DataFetchingEnvironment

Filter predicates were formerly a simple `Predicates`. They now use a dedicated interface which marks them as `ExtensionPoints` and they act as a `BiPredicate`.

[CMS-19769]

Headless Server Deprecations for CoreMedia CMCC 11

These previously announced deprecations were carried out with the major release of CoreMedia CMCC 11:

- Site Query: The deprecated parameter 'id' was removed from content-schema in favor of the new parameter 'siteId'
- `JsonPreviewUrlController` / Preview Url Endpoint (`/previewUrl`) was removed. `JsonPreviewUrlController` calculated the url for the json preview in studio. The url calculation was moved to studio server, where it is now part of the multi preview configuration.
- Method removed from `SolrQueryBuilder`: `getSortFieldMapping()` was removed
- Method removed from `SearchQueryHelper`: `validFromQuery()` was removed and replaced by `validFromPastToValueQuery()`
- Method removed from `SearchQueryHelper`: `validToQuery()` was removed and replaced by `validFromValueToFutureQuery()`
- Method removed from `ContentRoot`: `getSite(DataFetchingEnvironment environment)` was removed and replaced by `ByPathAdapter.getSite(@Nullable String siteId, @Nullable String rootSegment)`
- Config Property Removed: 'caas.cacheCapacities' - Cache capacities can be configured instead via `com.coremedia.cache.config.CacheConfigurationProperties`

[CMS-19685]

New Feature: Plugin Extension Point FilterPredicate

With the new extension point `FilterPredicate` it is possible to add filter predicate via the means of a plugin. This change is considered breaking, as customers/implementers have to use the new interface `FilterPredicate` instead of the more generic `Predicate<Object>`. `FilterPredicate` is just a marker interface, so the necessary changes are minimal.

[CMS-19641]

AbstractDynamicListAdapter#getItems and SearchServiceAdapter#search now return DataFetcherResults

`AbstractDynamicListAdapter#getItems` and `SearchServiceAdapter#search` now return a `DataFetcherResult`, which wraps data and error information. This enables a uniform way to transfer information about errors from the backend to the client via the `errors` field in the GraphQL response.

[CMS-19428]

Removal: Deprecated Headless API for Commerce Entities

The old commerce schema extension of the CoreMedia Headless Server was deprecated in 2101 and is now removed.

It is replaced by a new augmentation extension. Pure commerce data are no longer mixed with content in a single GraphQL schema. The CoreMedia Headless Server provides only content and augmentation data now. For more details, read the Release Notes "New Headless Commerce Schema" in Release 2104.1 [CMS-19117].

[CMS-19322]

Removal of ViewBySiteDataFetchingEnvironment, Public API Changes of RemoteLinkWiringFactory and RemoteLinkDataFetcher

The class `ViewBySiteDataFetchingEnvironment` has been removed. Use builder of `DataFetchingEnvironmentImpl.newDataFetchingEnvironment()` instead. The `RemoteLinkWiringFactory` and the `RemoteLinkDataFetcher` take a `remoteLinkDataLoader` as constructor argument. The method `CaasWebConfig#getForwardHeaderMap` has been removed. Retrieve the map from the environment context instead.

[CMS-19155]

Addition: New RichText Type to Encapsulate the Different Representations of Richtext

Introduced a new `RichText` type in the schema to encapsulate the different representations of `RichTexts`, like text as tree and linked contents in `RichText`. The old fields are deprecated and moved into the new type. The field `detailTextLegacy` and `teaserTextLegacy` were introduced to provide easy access to the former behavior of the `detailText` and `teaserText` fields.

[CMS-19107]

Changed Configuration Properties

The following configuration properties have changed and are now defined in the class `CaasSearchConfigurationProperties`:

- `caas.graphql.max-search-limit` changed to `caas.search.max-search-limit`
- `caas.querylist-search-cache-for-seconds` changed to `caas.search.cache.querylist-search-cache-for-seconds`
- `caas.solr.collection` changed to `caas.search.solr.collection`

Additionally the class `RemoteServiceConfiguration` was renamed to `RemoteServiceConfigurationProperties`.

[CMS-19076]

Ordered Content Schema

The content-schema file was ordered alphabetically by types, in order to raise the readability of the schema. Since the changed order does not change the code, this change does not break directly. However, customers who changed the content schema directly in that file [Blueprint], will most likely encounter a merge conflict.

[CMS-19071]

Site Query by Root Segment

Starting with 2104.1, the site query now additionally supports 'rootSegment' as a query parameter as an alternative to 'siteId'. The new feature is considered breaking, if the pageByPathAdapter was used! PageByPathAdapter was renamed to ByPathAdapter and the implementation to fetch a site object was moved there. The old implementation in ContentRoot was deprecated.

[CMS-18930]

Loading of Persisted Queries from Plugins

Headless server supports loading resource files containing persisted query definitions from the content of plugins.

By loading persisted queries from plugins rather than from within the Headless Server Docker image, the build, deployment and release management of the core application is decoupled from the development of persisted query resource files.

Loading persisted queries from plugins is optional. The deployment within the Headless Server App remains unchanged.

With CM11 the configuration prefix of 'caas-rest' was refactored to 'caas.rest' in order to meet standard naming conventions.

[CMS-18315]

Content Types in Search are Now Validated

While searching, non existent content types were simply ignored, leading to the situation, that the search wasn't restricted to any content type at all. As this behavior led to bigger result set than expected and may hide bugs on the client side, a content type validation was introduced. Whenever an invalid, non existent content type is given, headless will respond with one GraphQL error message per invalid content type.

This change is classified 'breaking', as a client might have used invalid content types without recognizing them as errors. If this is this case, the new behavior will break the client!

[CMS-17298]

CMProductTeaser change in headless

CMProductTeaser#teaserTarget and CMProductTeaser#teaserTargets always return null since only the CMProductTeaser#productRef should only be used. The teaser target can not be set in Studio anyways.

No action should be required.

[CMS-20193]

Headless Commerce: Remove Autocompletion for Breadcrumb Parameter

The breadcrumb parameter is not completed automatically anymore, if the category "root " is missing. Make sure that the breadcrumb parameter contains all categories from leaf to root category, including the root category itself.

Please review usages of `content#`

- `categoryAugmentationBySite`
- `categoryAugmentationBySegment`
- `productAugmentationBySite`
- `productAugmentationBySegment`

[CMS-20085]

JsonPreviewClient was refactored to Spring AutoConfiguration

The `JsonPreviewClient` was refactored to use Spring AutoConfiguration. As a consequence class `com.coremedia.blueprint.headlessserver.JsonPreviewClientConfig` was removed from blueprint. The removal is breaking, if `JsonPreviewClientConfig` was adapted by developers. If not, the change is non-breaking, as `JsonPreviewClientConfig` only contained a component scan which was replaced by the new AutoConfiguration.

[CMS-19941]

Removal: ViewMapperConfig Class

The Spring Configuration class `com.coremedia.caas.web.ViewMapperConfig` was removed from from public API. The change didn't affect any configuration properties. The config was meant to expose the existing view mappings during runtime, as the mapping may change during runtime by editing, removing or deleting the homepage of a site. As a configuration class is not meant to expose dynamic runtime data, the class was refactored. The mappings can now be acquired via an actuator at `/actuator/siteRestrictedEndpoints`.

[CMS-19885]

Aligned Behavior of StructAdapter to Same as in SettingsAdapter

When querying for a settings path via `StructAdapter`, the behavior varied from the one implemented in `SettingsAdapter`. The behavior of the `StructAdapter` was now aligned to the behavior of the `SettingsAdapter`. The documentation was updated accordingly.

[CMS-19452]

Default Java heap for Headless Server reset to 1 GB in Dockerfile

This change only affects customers coming from CoreMedia versions 2101.1, 2101.2 and 2101.3 who use the Docker deployment. The default Java heap for the Headless Server

has been reset from 2048m to 1024m in the Dockerfile of the Headless Server. The Java heap can, but does not have to be increased for performance optimization.

[CMS-19210]

Headless Server Schema Changes

Replace the usage of `Locale::toString` with `Locale::toLanguageTag` for the locale fields. This changes the behavior of the locale-field of the Site and CMLocalized types by changing the format of the locale field. Additionally removing the deprecated "languageId" field from the Headless Server schema.

[CMS-19096]

Support ContentBlobLinkComposer in RichText conversion

Added the `ContentBlobLinkAdapterFactory` to the `EvaluationContext` to be able to use the `ContentBlobLinkComposer` in `RichTextConversion`

[CMS-19055]

5.5.10 Miscellaneous

The site-manager container image is now also build with google jib

Because of this change, the convenience feature, that the zip will be configured at startup has been removed.

[CMS-20332]

Support for IBM DB2 has been removed

Support for IBM DB2 has been removed

[CMS-20120]

Removal: Deprecated Private API BlobHelper Class

The private API class `com.coremedia.cap.transform.undoc.BlobHelper` has been removed.

You may be affected, if you didn't incorporate changes to Blueprint which came with 1907.1 in these classes:

- `com.coremedia.blueprint.cae.handlers.CapBlobHandler`
- `com.coremedia.blueprint.cae.handlers.TransformedBlobHandler`
- `com.coremedia.blueprint.assets.cae.DownloadCollectionZipCacheKey`

They made use of this private API prior to 1907.1.

For a replacement, see the new implementations as they ship with CoreMedia Blueprint.

[CMS-20049]

Removal: Deprecated Editorial Comments cache properties `editorial.comments.data.cache.*` Properties

The deprecated configuration properties `editorial.comments.data.cache.*` have been removed.

- `editorial.comments.data.cache.comments-cache-size` is now `cache.capacities.com.coremedia.editorial.comments.data.editorial.comments`
- `editorial.comments.data.cache.qualifiers-cache-size` is now `cache.capacities.com.coremedia.editorial.comments.data.editorial.comments.qualifiers`
- `editorial.comments.data.cache.threads-cache-size` is now `cache.capacities.com.coremedia.editorial.comments.data.editorial.comments.commentThreads`

[CMS-20047]

Removal: Deprecated SettingsPageResourceBundleFactory Class

The deprecated class `com.coremedia.blueprint.cae.web.i18n.SettingsPageResourceBundleFactory` has been deleted. If you still like it, keep it in your project code. As alternative, it is recommended moving resource bundles to `resourceBundles`, and using `LinklistPageResourceBundleFactory` instead.

[CMS-19999]

Removal: Deprecated ThemeImporterImpl#legacyExternalLinkToDocumentName Method

The deprecated method `com.coremedia.blueprint.themeimporter.ThemeImporterImpl#legacyExternalLinkToDocumentName` has been deleted. If you still like it, keep it in your project code. Otherwise, it is recommended using `stringToDocumentName` instead.

[CMS-19995]

Removal: Deprecated UserProvider2 Interface

The interface `UserProvider2` has been deleted. If you are still using it in your project code, simply switch to `UserProvider`.

[CMS-19988]

Removal: Deprecated Legacy RuleProviders Classes

`com.coremedia.blueprint.base.links.impl.PropertiesRuleProvider` and `com.coremedia.blueprint.base.links.impl.SitesRuleProvider` have been deleted from the Java API, incl. the `bpbase-links-legacy.xml` configuration file. You probably do not use the classes directly in your project code. Instead, check if you still import the legacy configuration file in the

`blueprint-links.xml` file in the `cae-base-lib` module, and eventually migrate to the `AbsoluteUrlPrefixRuleProvider`.

[CMS-19987]

Removal: Deprecated `LdapMember#getOrganizationalUnit` Method

We have finally deleted `LdapMember#getOrganizationalUnit`, which has never been used. If you have implemented the `LdapMember` interface, you can delete the method. You must at least delete the `@Override` annotation.

We also deleted `LdapUserProviderConfigurationProperties#supportMemberOus`, which was used to activate `LdapMember#getOrganizationalUnit` (and which was `false` by default).

With the deletion of `LdapMember#getOrganizationalUnit`, the interface `OrganizationalUnit` became obsolete and has therefore been deprecated.

[CMS-19986]

Removal: `management.health.contentRepository.enabled` Property

The deprecated application property `management.health.contentRepository.enabled` has been removed. Please use its replacement `management.health.uapiConnection.enabled`.

[CMS-19962]

Migrated Blueprint Content Hub Adapters to Plugins

The extensions for the Content Hub Adapters

- `content-hub-adapter-rss`
- `content-hub-adapter-youtube`

have been removed from the Blueprint.

These adapters are now released as separate plugins which are available on GitHub.

- <https://github.com/CoreMedia/content-hub-adapter-rss>
- <https://github.com/CoreMedia/content-hub-adapter-youtube>

By default, the plugins are bundled with the blueprint.

For details on bundled plugins see `workspace-configuration/plugins/README.md` in the Blueprint workspace and section "Plugin Descriptors and Bundled Plugins" in the Blueprint Developer Manual.

To remove these bundled plugins, go to `workspace-configuration/plugins/` remove the entries from the file `plugin-descriptors.json` and run `mvn generate-resources`.

If you have customized the adapters, you can keep your customized extensions, but must remove the bundled plugins as described above. Nevertheless, in the long run, you should consider migrating your extensions to plugins as well.

[CMS-19911]

Dropped supported deprecated databases, servers, and browsers

With CMCC v11 the following deprecated databases and servers are not supported anymore: IBM DB2, MySQL 5.7, Postgres 9.6, MongoDB 3.6, and Microsoft Windows Server 2016 Standard.

Microsoft Internet Explorer 11 is also not supported in CoreMedia Studio anymore.

[CMS-19867]

New API For Custom Workflow Extension Point

Accompanying the new Workflow app, there is a new API to add custom workflows. The former API was targeted specifically at the Control Room in the main Studio. The new API can be used to customize both the main Studio and the Workflow App.

Note that the old API is still in place. As workflows continue to be started in the Control Room, the old API can still be used to customize the start-workflow windows.

The new API is designed to be more declarative and independent from ExtJs. For more details, consult the chapter "Customizing Workflow UI" in the Studio Developer Manual.

[CMS-19858]

Removal: Corporate Extension Modules for Content Server

The [content] server modules for the *corporate* extension were empty and have been removed:

- `com.coremedia.blueprint:corporate-server`
- `com.coremedia.blueprint:content-server.corporate`

See [Blueprint Developer Manual - Project Extensions](#) if You need to re-add the modules for customization purposes.

[CMS-19851]

Removal: Unused class SiteToStoreContextCacheKeyWithTimeout

The deprecated class `com.coremedia.blueprint.base.livecontext.ecommerce.common.SiteToStoreContextCacheKeyWithTimeout` was part of the legacy Blueprint commerce integration and has been removed in favour of the Commerce Hub integration.

[CMS-19850]

New API for Content Type Localization

To support content type localization in both the Main Studio App and the Workflow App (and further apps in the future), we fundamentally changed the API to localize doctypes. One particular change is the direct support for SVG icons.

For details on the new API, we refer to the section "Content Type Model" of the Studio Developer Manual.

For details on how to migrate existing content type localizations to the new API, we refer to the section XY of the CoreMedia Content Cloud v11 Upgrade Guide.

[CMS-19816]

Removal: Old live context settings struct

The old 'livecontext.store.config' StructProperty was removed for all commerce Hub enabled sites, tests and demo data. Please use only the new struct below the 'commerce' struct. For more details see JavaDoc for CommerceSettings.java. In addition LiveContextCloneSitePostProcessor.java and it's usages were removed completely, the CommerceCloneSitePostProcessor.java is used instead. Please note: the cms catalog is not affected by this refactoring as it's not a commerce hub enabled site.

[CMS-19396]

Removal: Deprecated API of TransformImageService

The following methods have been deleted from the Java API:

- `com.coremedia.cap.transform.TransformImageService#transformWithDimensions(Content content, Blob data, TransformedBlob operationsForTransformation, String variantName, String extension, Integer width, Integer height)` - Use `{com.coremedia.cap.transform.TransformImageService#transformWithDimensions(Content, String, String, Integer, Integer)}` instead.
- `com.coremedia.cap.transform.TransformImageService#getTransformationOperations(Content picture, String property, Map<String,String> transformations)` - Replace with `com.coremedia.cap.transform.TransformImageService#getTransformationOperations(Content, String)`, which also resolves the given transformation with the help of `ContentOperationsResolver`.

[CMS-19254]

Logging Standardization

The logging setup has changed to implement a more standardized setup. This includes:

- the removal of any thirdparty logback appenders to push logs to a log aggregation like elasticsearch.
- the removal of the Spring-Boot `elk` profile to activate elastic log aggregation

- the removal of the `com.coremedia.blueprint:logging-config` maven module
- the removal of any `logback-spring.xml` in favor of Spring-Boot standard setup
- the removal of the Studio Console logfile endpoint
- a simpler default logging pattern

removal of logback appenders

Instead of using this approach please use one of the many log shippers available to send logfiles to an elastic stack.

removal of logback-spring.xml files

If you need to customize your logging you can read `logback-spring.xml` files to the `*-app/src/main/resources` folder and follow the standard customization steps described in the Spring Boot documentation.

removal of Studio Console endpoint

The endpoint was intended to be used similar to the logfile endpoint of the standard Spring Boot actuators. To view the `studio-console` logger, please use standard log queries in your log aggregation stack of your choice to separate the `studio-console` logger from the studio-server logs.

standard console logging pattern

For an easier integration with log shippers, the logging pattern has been changed:

- any coloring has been removed for console and file logging except for `local` profiles, when the application is started using the IDE or the `spring-boot-maven-plugin`.
- the timestamp has been removed for standard console logging. The container logging always contains the timestamp and the shippers parse this timestamp anyway.
- the pattern has been reordered so that the message is the last entry. This simplifies the handling of multiline strings.

The new logging pattern for the file logger will be:

```
%d{yyyy-MM-dd HH:mm:ss} %-7([%level]) \\(%thread\\) %logger [%X{tenant}] -
```

```
%message%n
```

The new logging pattern for the console logger will be:

```
%-7([%level]) \\\(%thread\\) %logger [%X{tenant}] - %message%n
```

If you want to keep the old pattern, set the following property respectively environment variable `logging.pattern.file`, `LOGGING_PATTERN_FILE` to

```
%d{yyyy-MM-dd HH:mm:ss} %-7([%level]) %logger [%X{tenant}] - %message  
\\(%thread\\)%n
```

For the console logger set `logging.pattern.console` or `LOGGING_PATTERN_CONSOLE` to

```
%clr(%d{yyyy-MM-dd HH:mm:ss} -){faint} %clr(%7([%level])) %clr(%logger){cyan}  
[%X{tenant}] %clr(-){faint} %message \\\(%thread\\)%n
```

[CMS-19224]

Removal: Optimizely Extension

The extension `Optimizely` has been removed from applications `CAE` and `Studio-Client`. The settings starting with "`optimizely.`" have no effect anymore and can be removed from custom content. Additionally the helper class `ExtensionAspectUtil.java` has been removed.

[CMS-19060]

New Blueprint Parent

A new parent POM `cms-blueprint-parent` has been introduced and is used as (transitive) parent for every Blueprint module. It contains basic plugin management, especially for the install- and deploy-plugin for which it is important that all modules use the same versions. If you need a custom parent in your Blueprint, you can simply search and replace our parent element. You could use the `cms-blueprint-parent` as parent of your custom parent, or integrate the pluginManagement as necessary.

[CMS-18934]

Replaced deprecated org.junit.Assert.assertThat

All remaining occurrences of the deprecated `org.junit.Assert.assertThat` have been replaced by the equivalent `org.hamcrest.MatcherAssert.assertThat`.

[CMS-18795]

Replaced Guava Function and co. with JDK counterparts

Guava Optional, Function, Predicate, Supplier and most Guava utilities from `com.google.common.collect` have been replaced with their JDK counterparts. Most of these Guava features are available in the JDK by now, so it is unnecessary to bother with a third-party library. Although it is not required, we would recommend that you also refactor these Guava usages in your code for consistency.

[CMS-18659]

Removal: Deprecated Consumer, Function, Predicate Classes

The classes `com.coremedia.common.util.Consumer`, `com.coremedia.common.util.Function` and `com.coremedia.common.util.Predicate` have been deleted from the Java API. All usages have been replaced with the according JDK classes of package `java.util.function`.

The APIs of the `Consumer` and the `Function` classes are identical. If you have implemented own consumers or functions, you only have to change the `implements` clauses from `com.coremedia.common.util.Consumer` to `java.util.function.Consumer`. If you have own predicates, you must change the `implements` clause and rename the `include` method to `test`.

The `Functions` and `Consumers` utility classes from the same package have also been deleted, since they make no sense without the actual classes, and were trivial anyway. The `Predicates` class still exists, but has been deprecated and usages can be replaced as described in its API documentation.

[CMS-18658]

Implementations of Comparable: Added Type Arguments

For several parts of the public API the implementation of the `Comparable` interface now uses correct generic type arguments. As part of this change, following interfaces/classes have been updated by adding type parameter.

- `com.coremedia.cap.user.Group`
- `com.coremedia.cap.user.User`
- `com.coremedia.cap.workflow.Process`
- `com.coremedia.cap.workflow.ProcessDefinition`
- `com.coremedia.cap.workflow.Task`
- `com.coremedia.cap.workflow.TaskDefinition`
- `com.coremedia.workflow.WfValue`
- `com.coremedia.workflow.common.expressions.Aggregate`

[CMS-17548]

Consolidate Blueprint groupIds and versions

The Blueprint now uses the groupId `com.coremedia.blueprint` for every module and Blueprint modules are now always referenced by using `$(project.version)`. The following groupIds are no longer used:

- `com.coremedia.blueprint.boot`
- `com.coremedia.blueprint.docker`
- `com.coremedia.blueprint.content-hub`

The following properties have been removed:

- `blueprint.groupId`
- `blueprint.version`
- `blueprint.boot.groupId`
- `blueprint.boot.version`

This makes it easier to replace the coremedia groupId with a custom one and allows for setting the version via `maven-release-plugin` or `versions-maven-plugin` which should also make it easier to create releases of your blueprint.

[CMS-17559]

6. Support

Support, Newsgroups

Online Support

Use our Online Support to submit a support ticket, track your submitted tickets or receive access to our forums. You can access our Online Support at:

<http://support.coremedia.com>

Access to online services

Partners and Support customers will need to register an account with CoreMedia in order to access the Online Support. Just send us an email and we will send you your login details:

support@coremedia.com

If you have any other questions or comments, please contact:

CoreMedia GmbH

Ludwig-Erhard-Straße 18

20459 Hamburg

Phone: +49 .40 .32 55 87 .777 Fax: .999

www.coremedia.com

Our Support employees are available to take your support call weekdays between 9 a.m. and 6 p.m.

Our manuals are continuously reviewed and revised in order to reflect new developments and insights gained from the day-to-day use of CoreMedia systems.

Documentation

If you have comments or questions about our manuals,

- please email your inquiry to documentation@coremedia.com
- or fax it to 040-325587 999.

Please provide all necessary information in your support request to ensure that our Support team can efficiently help you with your issue.

CoreMedia systems are distributed systems that have a rather complex structure. This includes databases, hardware, operating systems, drivers, virtual machines, class libraries, customized code etc. in many different combinations. That's why CoreMedia needs detailed information about the environment for a support case. In order to track down your problem, provide the following information:

- Which CoreMedia component(s) did the problem occur with (incl. release number)?
- Which database is in use (version, drivers)?
- Which operating system(s) is/are in use?
- Which Java environment is in use?
- Which customizations have been implemented?
- a full description of the problem (as detailed as possible)
- Can the error be reproduced? If yes, give a description please.
- How are the security settings (firewall)?

In addition, log files are the most valuable source of information.

To put it in a nutshell, CoreMedia needs:

1. a person in charge (ideally, the CoreMedia system administrator)
2. extensive and sufficient system specifications
3. detailed error description
4. log files for the affected component(s)
5. if required, system files

An essential feature for the CoreMedia system administration is the output log of Java processes and CoreMedia components. They're often the only source of information for error tracking and solving. All protocolling services should run at the highest log level that is possible in the system context. For a fast breakdown, you should be logging at debug level. The location where component log output is written is specified in its `<appName>-logback.xml` file.

Which Log File?

Mostly at least two CoreMedia components are involved in errors. In most cases, the *Content Server* log files together with the log file from the client. If you are able locate the problem exactly, solving the problem becomes much easier.

Where do I Find the Log Files?

By default, log files can be found in the CoreMedia component's installation directory in `/var/logs` or for web applications in the `logs/` directory of the servlet container. See the CoreMedia Deployment Manual, "Service and Logfiles Names" chapter for the default names of the logfiles and the Operations Basics Manual, "Logging" chapter for more details.

Support request

Support checklist

Log files