CoreMedia

Release Notes

CoreMedia Content Cloud - v13

Copyright CoreMedia GmbH © 2026

CoreMedia GmbH

Altes Klöpperhaus, 5. OG

Rödingsmarkt 9

20459 Hamburg

International

Germany

Licenses and Trademarks

January 28, 2026 (Release 13.2512.0)

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-13/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 Command line entries Parameter and values	Courier new	`cm systeminfo`
Menu names and entries	Bold, linked with |	Open the menu entry **Format|Normal**
Field names CoreMedia Components	Italic	Enter in the field *Heading* The *CoreMedia Component*
Entries	In quotation marks	Enter "On"
(Simultaneously) pressed keys	Bracketed in "<>", linked with "+"	Press the keys <Ctrl>+<A>
Emphasis	Italic	It is *not* 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:

Release 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.

Available releases

The release overview site at https://releases.coremedia.com/cmcc-13/overview/archive.html provides an overview of all available release versions of *CoreMedia Content Cloud*.

Helpcenter

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.

Module organization

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
  Search Engine
  Command Line Tools
- **CoreMedia Studio:** All changes related to *CoreMedia Studio*
- **CoreMedia Elastic Social:** All changes related to Elastic Social
- **CoreMedia Native Personalization:** All changes related to Native Personalization

- **CoreMedia Frontend:** All changes related to the CoreMedia Frontend Workspace.

  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 2512.0.0

The release contains:

- Breaking changes: 84
- Changes and improvements: 287
- Bugfixes: 270

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.

Release date: 2025-12-11

Breaking Changes

- **CMS-29380:** Minor API Change in WorkflowObject
- **CMS-29293:** Removed Built-in Search and Replace Workflows
- **CMS-28803:** Configuration for Command Line Tool "publishall" Changed
- **CMS-28599:** Feeder caches results of Tika text extraction
- **CMS-28490:** Support for personalization variants
- **CMS-28413:** CAE Feeder Renamed Cache Class for Internal Caching
- **CMS-28375:** Removed Deprecated Method in AssetDocumentBlobFeedablePopulator
- **CMS-28374:** Removed Deprecated Classes ValidFromPropertyConverter / ValidToPropertyConverter
- **CMS-27992:** Removal: Deprecated Java API Package com.coremedia.cap.server.search
- **CMS-27871:** Removal of XML Importer
- **CMS-27456:** Hardened Server- and XLIFF-Export for Invalid XML Control Characters
- **CMS-27398:** CAE Feeder Spring Configuration Changes and Deprecations
- **CMS-25885:** Removed cm groovysh
- **CMS-23931:** Deprecated Workflow JAR Upload Disabled by Default
- **CMS-23177:** Removed Obsolete Property sql.store.login-user-name
- **CMS-18857:** Content Server Database Change

- **CMS-29071:** Cache Control config is mapping fragment urls by default
- **CMS-28410:** Image Transformation Operation: @Param.alias Changed to String-Array
- **CMS-27565:** Removed Deprecated Template API
- **CMS-27562:** CAE: Deprecated public api removed
- **CMS-27279:** Replaced FreeMarker XML Spring Configuration with FreeMarker Spring Boot AutoConfiguration
- **CMS-22262:** Profile "turbojpeg" Removed
- **CMS-29309:** Public API changes for Localization / Severity
- **CMS-29268:** Third party updates
- **CMS-28797:** Content Variants Always Open in the Same Studio Tab And Opening Studio Tabs is Asynchronous
- **CMS-28659:** SearchContentLinkSuggester deprecated API removed.
- **CMS-28594:** PNPM: minimumReleaseAge is now set to one day by default
- **CMS-28547:** Node.js 24 and PNPM 10.24.0 Upgrade
- **CMS-28526:** Removed Animations
- **CMS-28434:** Removed Tab-Expand
- **CMS-28424:** Studio: Deprecated `SearchService` replaced with `SearchService2`
- **CMS-28267:** CKEditor5 has been updated to version 46.1.1
- **CMS-27991:** Studio-Client packages for build-config, eslint-config and i18n-models now have their own release cycle
- **CMS-27980:** Studio Client Workspace now employs the PNPM Catalog Protocol
- **CMS-27952:** Deprecation: @coremedia/studio-client.client-core-impl and @coremedia/studio-client.cap-rest-client-impl
- **CMS-27789:** Include Updated Referenced Content in Publication Workflows
- **CMS-25612:** Public API changed
- **CMS-25598:** Upgraded eslint to v9
- **CMS-21880:** Old BlueprintDocumentTypes_properties is gone
- **CMS-6803:** Removed deprecated APIs from modules cap-rest-service and bpbase-studio-rest
- **CMS-28130:** Upgraded to Spring Boot 4 / Spring 7 baseline
- **CMS-27846:** Support and Require Maven 3.9.11
- **CMS-29269:** Updated Third Party Dependencies

- **CMS-29265:** Frontend Workspace now employs the PNPM Catalog Protocol
- **CMS-25597:** Upgraded eslint to v9
- **CMS-27827:** compose-wait command-line tool has been removed from the base image
- **CMS-22475:** confd has been removed from the java-application-base image
- **CMS-16467:** Switch studio-client and overview to CoreMedia vendored unprivileged nginx image.
- **CMS-27833:** Renamed grpc.stubs Properties
- **CMS-27122:** Commerce Hub gRPC Client Requires Proto Reflection V1
- **CMS-28677:** Headless Server: Deprecated content type 'application/graphql+json' not supported anymore
- **CMS-28460:** Headless now support image type transformation for original images (non-crop)
- **CMS-28253:** Headless Server Content-Schema Deprecations in CM13
- **CMS-27563:** Headless Server Java-API Deprecations in CM13
- **CMS-27336:** Changed Cache Control Defaults
- **CMS-25427:** Headless Server: Improved extensibility and flexibility of Blob delivery
- **CMS-29078:** Deprecation of Adaptive Personalization
- **CMS-28780:** Removed unused code from UriComponentsHelper
- **CMS-28763:** Adaptive Personalization (p13n) Extensions Disabled By Default
- **CMS-28689:** gRPC as an Alternative to CORBA Communication
- **CMS-28541:** CORS Configuration Updated
- **CMS-28515:** Removed redundant dependency management from blueprint parent POMs
- **CMS-28497:** Removed util class InputInput
- **CMS-28438:** Do not start MongoDB and Elastic Worker by default
- **CMS-28214:** Removed deprecated BaseFileSystemConfiguration from ContentHub API
- **CMS-28213:** Content hub adapters are no longer delivered as bundled plugins
- **CMS-28177:** CatalogPicture#getUrl() removed from template api
- **CMS-28146:** Removed Deprecated API: Predicates
- **CMS-28074:** Removal of Analytics Connectors
- **CMS-28072:** Elastic Social (es) Extensions Disabled By Default

- **CMS-27936:** Support for MySQL 8.0 and PostgreSQL 13 has been removed
- **CMS-27806:** The workflowconverter initscript has been removed from the workflow-server image
- **CMS-27753:** Confd has been removed from the management-tools image
- **CMS-27564:** Middle Tier: Removal of Deprecated API
- **CMS-27557:** Common Tier: Removal of Deprecated API
- **CMS-27540:** Third-Party Update: Apache Solr 9.10.0
- **CMS-27499:** Discontinue base image flavors
- **CMS-27495:** Switched to spring-grpc
- **CMS-27319:** Java 21 Support
- **CMS-27298:** Removal of deprecated Commerce Hub Data properties
- **CMS-27251:** Removed Deprecated API for Custom Proactive Engine Application
- **CMS-26030:** Studio-Client Docker build improvements
- **CMS-25905:** Support for Microsoft SQL Server (MSSQL) has been removed
- **CMS-25582:** Change Storage Layer of Collaborative Components to SQL

Changes and Improvements

- **CMS-29030:** JShell Enhancements
- **CMS-28582:** Third-Party Update: Apache Tika 3.2.3
- **CMS-28465:** Deprecated Content Feeder Configuration Properties
- **CMS-28431:** Third-Party Update: Apache Tika 3.2.2 and Transitive Dependencies
- **CMS-27872:** Plugins can Contribute Document Type Aspects
- **CMS-27700:** Updated Project Dependencies
- **CMS-27698:** Session Check for Blob Download from Content Server Can Be Disabled
- **CMS-25999:** MongoDB update to version 8.0
- **CMS-17555:** Add Checkbox To Publication Workflow Window To Include Referenced Updated Content
- **CMS-28554:** Fixed SSRF vulnaribility in CodeResourceHandler
- **CMS-29266:** Coremedia npm repository changed
- **CMS-28961:** Removed Outdated Documentation about Teaser Text Delegation
- **CMS-28960:** Pendo plugin no longer bundled by default

- **CMS-28655:** Increase Tool Tip Delay
- **CMS-28416:** Improved Shortcuts Description
- **CMS-28088:** New icon design
- **CMS-27970:** Character Counter for TextFields
- **CMS-22696:** Fixed Single-Selection Mode for the Taxonomy Chooser
- **CMS-28322:** For kubernetes test system deployments a new overview app will be installed
- **CMS-25927:** Cleaned up dependency-management
- **CMS-28178:** Dedicated Cache Class for Commerce Objects
- **CMS-28152:** Faceted search query returns facet content
- **CMS-28142:** Taxonomy localization
- **CMS-28987:** Fix default for KIO Playbook Github Token
- **CMS-28786:** Resilience4j Update
- **CMS-28721:** Update of JUnit Jupiter
- **CMS-28559:** Updated MySQL Driver
- **CMS-28032:** Configurable Cache Duration for Outdated Transformed Blobs
- **CMS-28006:** JUnit Jupiter: Updated to 5.13.2
- **CMS-27967:** Updated Hibernate to 7.1
- **CMS-27839:** Third-Party Update: commons-io 2.19.0
- **CMS-27810:** New methods Content#getLatestVersion, Content#getLatestOrWorkingVersion, Content#getCheckedInOrWorkingVersion and Content#getCheckedOutOrWorkingVersion
- **CMS-27692:** Removed unused legacy CORS configuration properties
- **CMS-27505:** Third-Party Update: Checker Framework
- **CMS-27460:** Third-Party Update: Eclipse Glassfish CORBA ORB
- **CMS-27281:** The git-commit-id-plugin has been removed from the maven build system
- **CMS-27258:** jshell has been added to all java-application-base image flavors
- **CMS-25935:** a new maven profile to build multi-arch images has been added
- **CMS-25844:** The cm based command-line tools have been moved to the corresponding server application images
- **CMS-23359:** LdapUserProvider Connection Improvements

Bugfixes

- **CMS-28851:** Workflow Action Exception Logging
- **CMS-28696:** XLIFF-Import: Relaxed xlink:href and xlink:role Validation
- **CMS-27908:** Deprecated Remote Client Actions
- **CMS-27798:** Increased Maximum Content-Type Name Lengths
- **CMS-25759:** Extended Output of cm dump Tool for Workflow Processes
- **CMS-25566:** Fixed Bug In Integrated Workflow Conversion With PostgreSQL
- **CMS-23988:** Archive Broken Workflow Definitions
- **CMS-16480:** Optionally Pass Authentication Data for Content Server's HTTP Endpoints in Request Header Fields
- **CMS-29178:** Configured Forced Response Character Encoding for CAE
- **CMS-28937:** Changed Themes Target Folder Name
- **CMS-28897:** Correction of CAE and HLS Cache Header max-age
- **CMS-28288:** Fixed Parameter Resolution of Dynamic Includes
- **CMS-28160:** Fixed JMX issue with CAE startup
- **CMS-27938:** Default configuration of cache-control headers fixed
- **CMS-16789:** Removed duplicate of `userVariant`
- **CMS-29040:** Improved Shortcuts Description
- **CMS-29037:** Improved Dialog Accessibility
- **CMS-28912:** Fixed an issue that time zone names were not propertly localized in studio-client
- **CMS-28699:** Fixed Workflow Sidebar Footer
- **CMS-28417:** Fixed Annotated LinkList Textfields
- **CMS-28369:** Fixes: Link Balloon closes on content with multiple richtext editors when opening the library/control room
- **CMS-28259:** Fixed Miscellaneous Aria Roles and Labels
- **CMS-28258:** Repository Tree Accessibility
- **CMS-28257:** Fixed Widget Button Tooltips
- **CMS-28254:** Fixed Miscellaneous Aria Roles and Labels
- **CMS-28252:** Fixed HTML "lang" Attribute
- **CMS-28250:** Tooltips are now activated on Focus
- **CMS-28243:** Welcome Widget is now accessible

- **CMS-28239:** Fixed Library Column Accessibility
- **CMS-28238:** Fixed Library Context Menu Accessibility
- **CMS-28133:** Added Missing QuickCreate Parameters
- **CMS-27993:** Image Editor: wrongly sorted image transformation strings
- **CMS-27643:** Skip unnecessary save action
- **CMS-27485:** Fixed: CKEditor drag&drop imports contents from other systems (e.g. Content-Hub, Commerce) to the site root folder.
- **CMS-27053:** BoundRadioGroup DefaultValue Warning
- **CMS-26035:** Fixed Miscellaneous Struct Editor Issues
- **CMS-25975:** Workflows in ControlRoom inbox are now correctly displayed
- **CMS-25609:** Fixed Tree Lazy Loading
- **CMS-24059:** Fixed Connection Recovering
- **CMS-21564:** Fixed: postMessage origin error
- **CMS-29318:** Users can no longer delete their own Home folder in Studio.
- **CMS-28777:** Replacing workflow start validators via Customize-Annotation leads to startup errors of Studio-Server
- **CMS-28685:** CKEditor5: Link-Balloon layout issue has been fixed
- **CMS-28336:** Fixed Search by Path in Studio Folder Chooser
- **CMS-27829:** Custom content creation paths from Editor Preferences fixed
- **CMS-27609:** Adjusted storage of editorial-comments timestamps to prevent issues with daylight saving times
- **CMS-25650:** Removed obsolete configuration from Studio Server
- **CMS-25282:** Fixed Order of Uploaded JSON Settings
- **CMS-28787:** Changed CaasConfig#remoteLinkDataLoader
- **CMS-28337:** Headless responds to HTTP-Head over REST now as in CM11
- **CMS-28203:** Richtext headings are resolved with inline handlers
- **CMS-29384:** Extended document types for XmlUapiConfig
- **CMS-29048:** Fixed locale-based search in the preferred site selector
- **CMS-28592:** Fix Blueprint SearchConstants Value
- **CMS-28574:** Studio Multi-Preview can Change or Disappear Depending on User Permissions
- **CMS-28505:** Fixed a Bug in Cap Repositories Spring Configuration
- **CMS-28100:** Category parameter is taken for augmented products
- **CMS-18211:** Changed the logic for the metric com.coremedia.Store.Connected

4.1.1 Breaking Changes

CoreMedia Core

Minor API Change in WorkflowObject

The return type of `com.coremedia.cap.workflow.WorkflowObject#getWarnings` has been weakened from `List<WorkflowException>` to `List<CapException>`. Due to a CMS internal generics inconsistency the result list of `getWarnings` may contain `CapException` objects even in older CMS versions. This fix does not change the behaviour, but only clarifies the type. Custom code may have to be adjusted accordingly.

(CMS-29380)

Removed Built-in Search and Replace Workflows

The built-in workflow definitions `global-search-replace.xml` and `markup-search-replace.xml` have been removed. These definitions were used with the Site Manager, which has already been removed in CMCC 12.

(CMS-29293)

Configuration for Command Line Tool "publishall" Changed

Command Line Tool *publishall* was previously configurable with non-option arguments: `cm publishall [ -a [ -cq <query> ] ] [ -t <threads> ] <cmsiorurl> <cmsuser> <cmspwd> <masteriorurl> <masteruser> <masterpwd>`. With the introduction of gRPC as the communication protocol and the need for additional parameters, the configuration was adjusted and unified. Instead of non-option arguments, now all configuration has to be given as options.

Follow Section “Configuration for Command Line Tool "publishall" Changed” [64] for upgrade information.

(CMS-28803)

Feeder caches results of Tika text extraction

Content Feeder and *CAE Feeder* now cache the results of Apache Tika text and metadata extraction for indexed blob values. This improves feeding performance, if content is re-indexed with unchanged blob value, or if the same blob value is used in multiple content items. The Blueprint configures a default cache capacity of estimated 10 MB heap memory using configuration property `cache.capacities.feeder.tika.heap` in `application.properties`. Because of that, Feeders require slightly more heap memory. Caching can be disabled by setting the cache capacity to 0.

(CMS-28599)

Support for personalization variants

In order to support personalization variants, a new meta-property `variant` was added to `Content` items. It is immutable once set during content creation. It is accessible through `isVariant()` and its inverse `isBaseline()`. For a baseline content item, multiple variants may be created. Variants are created with the same content type as their baseline. They are used for delivering A/B tests or specialized content for user segments.

Follow Section “Support for personalization variants” [66] for upgrade information.

(CMS-28490)

CAE Feeder Renamed Cache Class for Internal Caching

The *CAE Feeder* uses in-memory cache keys for caching fragment values returned by Java method `com.coremedia.cap.persistentcache.PersistentCache#getCached`. The cache class of these internally used cache keys has changed from the generic cache class `java.lang.Object` to the new cache class `com.coremedia.cap.persistentcache`. The default cache capacity of 10000 entries is the same as the Blueprint’s default for the previously used cache class. For details, see also section “Getting the Fragment Key Value from the Persistent Cache” in the *Search Manual*.

(CMS-28413)

Removed Deprecated Method in AssetDocumentBlobFeedablePopulator

Removed deprecated method `setMaxBytes` of Blueprint class `com.coremedia.blueprint.feeder.cae.assets.AssetDocumentBlobFeedablePopulator`. Use configuration property `feeder.blob.max-size` instead to limit feeding of blobs by their size.

(CMS-28375)

Removed Deprecated Classes ValidFromPropertyConverter / ValidToPropertyConverter

Removed deprecated Blueprint Java classes `ValidFromPropertyConverter` and `ValidToPropertyConverter` in package `com.coremedia.blueprint.caefeeder`.

(CMS-28374)

Removal: Deprecated Java API Package com.coremedia.cap.server.search

The following deprecated interfaces and classes of package `com.coremedia.cap.server.search` have been removed from the Java API:

- `com.coremedia.cap.server.search.SearchContext`
- `com.coremedia.cap.server.search.SearchException`
- `com.coremedia.cap.server.search.SearchService`
- `com.coremedia.cap.server.search.SearchServiceResult`

(CMS-27992)

Removal of XML Importer

The deprecated XML Importer has been removed. As a replacement, a custom Unified API client, the Ingest Service for CoreMedia Content Cloud Service or the serverimport tool should be used.

Follow Section “Removal of XML Importer” [68] for upgrade information.

(CMS-27871)

Hardened Server- and XLIFF-Export for Invalid XML Control Characters

The output of `cm serverexport` as well as the resulting XLIFF from XLIFF-Export have been hardened to not output invalid XML control characters. If these characters made it into the XML/XLIFF-files parsers fail reading them afterward.

Invalid control characters range from `0x0000` (NUL) to `0x001F` (Information Separator One (US - Unit Separator)) except for `0x0009` (Horizontal Tab), `0x000A` (Line Feed) and `0x000D` (Carriage Return). On export, these control characters are now removed, so that they don’t block subsequent import.

One exception to this rule: For attributes `name` and `folder` of a given exported content item, URL-encoding is applied to them now, just as it is already done for internal representation (in link lists as well as in links in Rich Text and Structs).

Specification Level 1: With this change we introduce a “specification level” explicitly noted in files created by `serverexport` such as:

```
<CMChannel
  specification="1"
  folder="/Sites/Chef%20Corp%2E/United%20States/English/Navigation"
  name="Chef%20Corp%2E%20USA%20Home%20Page"
  xmlns:cmexport="http://www.coremedia.com/2012/cmexport">
  <!-- ... -->
</CMChannel>
```

Follow Section “Hardened Server- and XLIFF-Export for Invalid XML Control Characters” [68] for upgrade information.

(CMS-27456)

CAE Feeder Spring Configuration Changes and Deprecations

The Blueprint CAE Feeder Spring configuration was changed to use Spring Java configuration classes instead of legacy Spring XML files. This makes it possible to use Java code to configure which content bean data should be indexed. Instead of configuring content bean classes and properties as literal string values, the configuration can now use more reliable Java code to refer to methods directly. To this end, new `addElement` methods have been added in class `com.coremedia.cap.feeder.bean.BeanFeedableMapping` while API for old-style Spring XML configuration has been deprecated.

Follow Section "CAE Feeder Spring Configuration Changes and Deprecations" [70] for upgrade information.

(CMS-27398)

Removed cm groovysh

We removed the command line tool `cm groovysh` in favor of the more mature solution for ad-hoc Unified API scripting and more using `cm jshell` that we ship since CMCC v10.2004.1.

Follow Section "Removed cm groovysh" [73] for upgrade information.

(CMS-25885)

Deprecated Workflow JAR Upload Disabled by Default

The deprecated feature to upload JAR files with classes, and to use classes from previously uploaded JAR files in workflow process definitions has been disabled by default, and may be removed in a future release. Required classes should be added to the Workflow Server classpath instead.

Follow Section "Deprecated Workflow JAR Upload Disabled by Default" [74] for upgrade information.

(CMS-23931)

Removed Obsolete Property sql.store.login-user-name

The property `sql.store.login-user-name` has become obsolete and has been removed. Just use `sql.store.user` instead.

Follow Section "Removed Obsolete Property sql.store.login-user-name" [74] for upgrade information.

(CMS-23177)

Content Server Database Change

When the *Content Server* starts the first time after upgrading, it will automatically remove the no longer needed foreign key constraint `fk_WfV_group` on column `groupValue_0` of table `WfVariables` for existing Oracle, MariaDB and MySQL databases.

Follow Section "Content Server Database Change" [74] for upgrade information.

(CMS-18857)

CoreMedia Content Application Engine

Cache Control config is mapping fragment urls by default

In the Cache Control configuration URL patterns can be used to define a certain cache behaviour. These URL patterns can be extended to include matrix parameters. Matrix parameters are crucial to map fragment URLs in a commerce-led scenario. The default was changed to support matrix parameters by default.

(CMS-29071)

Image Transformation Operation: @Param.alias Changed to String-Array

`@Param` (FQN: `com.coremedia.transform.dispatch.Param`) got its parameter `alias` changed from type `String` to `String[]`.

The new feature is used for a typo in the `brightness` operation parameter “amount” that was misspelled as “amout”. The definition is now rewritten to also still support “amout” if used in any stored transformation Strings:

```
@Operation(alias = "b")
void brightness(ImageTransformerState<Image> state,
               @Param(name = "amount", alias = {"a", "amout"}) double amount,

                @Param(name = "contrast", alias = "c", defaultValue = "1.0")
 double contrast);
```

Follow Section “Image Transformation Operation: @Param.alias Changed to String-Array” [75] for upgrade information.

(CMS-28410)

Removed Deprecated Template API

The following deprecated API and all their usages have been removed:

- com.coremedia.blueprint.common.layout.Container#getFlattenedItems
- com.coremedia.blueprint.common.contentbeans.CMMedia#getCaption

Along with this change, the content bean aspect API (package `com.coremedia.cae.aspect`) has been undeprecated.

The Blueprint content bean methods `getAspects` and `getAspectsByName` are no longer deprecated. All related deprecation annotations have been removed.

(CMS-27565)

CAE: Deprecated public api removed

The following deprecated API has been removed:

- `com.coremedia.cae.webflow.FlowRunner`

- `com.coremedia.blueprint.base.multisite.cae.SiteResolver#findSiteByPath`
- `com.coremedia.objectserver.web.ContentBlobHandlerBase`

Follow Section "CAE: Deprecated public api removed" [76] for upgrade information.

(CMS-27562)

Replaced FreeMarker XML Spring Configuration with FreeMarker Spring Boot AutoConfiguration

CoreMedia's XML Spring configuration for the FreeMarker integration has been replaced with Spring Boot's `@AutoConfiguration` for FreeMarker.

Follow Section "Replaced FreeMarker XML Spring Configuration with FreeMarker Spring Boot AutoConfiguration" [77] for upgrade information.

(CMS-27279)

Profile "turbojpeg" Removed

Profile *turbojpeg* has been removed from the Blueprint workspace. It was experimental in the first place and unsupported for several years. Built-in image transformation in CAE and Headless Server continues to work as before. If in need for high-performance image transformation, consider using CoreMedia's Image Transformation Service.

(CMS-22262)

CoreMedia Studio Client

Public API changes for Localization / Severity

The following types / consts / classes are now deprecated. You can find additional information about what to use instead in the code documentation:

- "Severity" of "@coremedia/studio-client.base-models"
- "ILocalization", "ILocalizationWithIcon", "ITypeLocalization", "IPropertyLocalization" and "IContentTypeLocalization" of "@coremedia/studio-client.cap-base-models"
- "IContentTypeLocalizationRegistry#getLocalizationByContentType" of "@coremedia/studio-client.cap-base-models"

Follow Section "Public API changes for Localization / Severity" [79] for upgrade information.

(CMS-29309)

Third party updates

The version of many third party dependency was updated to their latest minor/patch versioning.

Follow Section “Third party updates” [79] for upgrade information.

(CMS-29268)

Content Variants Always Open in the Same Studio Tab And Opening Studio Tabs is Asynchronous

We introduced the concept of certain "content item groups" that are always opened in the same Studio tab. For example. content items that are localized variants of each other (they have the same transitive master content) are now always opened in the same tab. It is no longer possible to open such variant-related content items in different tabs.

Follow Section “Content Variants Always Open in the Same Studio Tab And Opening Studio Tabs is Asynchronous” [80] for upgrade information.

(CMS-28797)

SearchContentLinkSuggester deprecated API removed.

The `SearchContentLinkSuggester` leaked private API. This API has now been removed. A replacement is not available as those functions were very specific and not mean to be part of the public API.

(CMS-28659)

PNPM: minimumReleaseAge is now set to one day by default

To reduce the risk of installing compromised packages, we now utilize the new minimumReleaseAge setting (introduced with 10.20.x) to delay the installation of newly published versions for 1 day. As in most cases, malicious releases are discovered and removed from the registry within an hour this is meant to protect us sufficiently while still being able to install fixes for other security issues in time.

Follow Section “PNPM: minimumReleaseAge is now set to one day by default” [81] for upgrade information.

(CMS-28594)

Node.js 24 and PNPM 10.24.0 Upgrade

- Node was upgraded to the latest LTS version 24.11.1
- PNPM has received a minor upgrade to 10.24.0

(CMS-28547)

Removed Animations

The .webm based Studio animations have been removed from the Studio and replaced by .svg files. All searches and empty states are affected by this, including the workflow app.

(CMS-28526)

Removed Tab-Expand

The configuration flag for the tab expand has been removed. All formular tabs are always shown now.

Follow Section “Removed Tab-Expand ” [81] for upgrade information.

(CMS-28434)

Studio: Deprecated `SearchService` replaced with `SearchService2`

The `SearchService` has been replaced with `SearchService2`. While the `SearchService` was based on Callbacks and Remote Beans the `SearchService2` is based on Promises.

(CMS-28424)

CKEditor5 has been updated to version 46.1.1

CKEditor5 has been updated to version 46.1.1 and may require an update of the `ckeditor5` dependency in custom plugins.

(CMS-28267)

Studio-Client packages for build-config, eslint-config and i18n-models now have their own release cycle

The versions of following packages are no longer matching the CoreMedia CMS Release version but have (similar to jangaroo packages) a dedicated release cycle:

- @coremedia/studio-client.build-config
- @coremedia/studio-client.eslint-config (old name: @coremedia/eslint-config-studio-client)
- @coremedia/studio-client.i18n-models

Follow Section “Studio-Client packages for build-config, eslint-config and i18n-models now have their own release cycle” [82] for upgrade information.

(CMS-27991)

Studio Client Workspace now employs the PNPM Catalog Protocol

The Studio Client Workspace now employs the usage of the PNPM catalog protocol for all dependencies in package.json of all packages. Using the catalog protocol makes it much easier and less error prone to maintain only one version per dependency in a workspace. For details refer to the original documentation at https://pnpm.io/catalogs

Follow Section “Studio Client Workspace now employs the PNPM Catalog Protocol” [82] for upgrade information.

(CMS-27980)

Deprecation: @coremedia/studio-client.client-core-impl and @coremedia/studio-client.cap-rest-client-impl

All modules exposed in @coremedia/studio-client.client-core-impl and @coremedia/studio-client.cap-rest-client-impl have been merged into @coremedia/studio-client.client-core and @coremedia/studio-client.cap-rest-client respectively keeping the path within the corresponding package intact. Please change usages (e.g. dependencies, imports) accordingly.

Examples

Follow Section “Deprecation: @coremedia/studio-client.client-core-impl and @coremedia/studio-client.cap-rest-client-impl” [83] for upgrade information.

(CMS-27952)

Include Updated Referenced Content in Publication Workflows

When starting a publication workflow in Studio there is now a checkbox to include updated referenced content - similar when starting a translation workflow. `PublicationWorkflowPlugin` and `DefaultStartPublicationWorkflowFormBase` also have the config parameter `hideDependentContentsStrategyChooser` which can be set to hide the checkbox.

(CMS-27789)

Public API changed

Public API Removal

@coremedia/studio-client.cap-rest-client

- CapList, CapListRepository and EncryptionService of are not public anymore.
- CapConnection: removed members getCapListRepository and getEncryptionService (were already marked as @private)

New Public API

Follow Section “Public API changed” [85] for upgrade information.

(CMS-25612)

Upgraded eslint to v9

Eslint was upgraded to v9.

Follow Section “Upgraded eslint to v9” [85] for upgrade information.

(CMS-25598)

Old BlueprintDocumentTypes_properties is gone

The old resource bundle BlueprintDocumentTypes_properties was used mostly for the localisation for the content types and their properties. As there are already a new mechanism for the content type localisation the old resource bundle is now removed from blueprint.

(CMS-21880)

CoreMedia Studio Server

Removed deprecated APIs from modules cap-rest-service and bpbase-studio-rest

The following deprecated API has been removed and may require adapting usages:

Module `cap-rest-service`:

- Added public API package `com.coremedia.rest.cap.content.search`.
- Removed unused `com.coremedia.rest.cap.content.UrlModifier`.
- Removed `CapObjectFormat` and `QueryUriResolver` from the public API package `com.coremedia.rest.cap.content.search` .

Module `bpbase-studio-rest`:

- Removed public API package `com.coremedia.blueprint.base.rest.convert`.

(CMS-6803)

CoreMedia Workspace

Upgraded to Spring Boot 4 / Spring 7 baseline

The Spring dependencies were upgraded to the Spring Boot 4 / Spring 7 baseline. With these upgrades, some further dependencies had to be upgraded as well.

Follow Section “ Upgraded to Spring Boot 4 / Spring 7 baseline” [88] for upgrade information.

(CMS-28130)

Support and Require Maven 3.9.11

Supported and Required Maven version has been updated to 3.9.11.

Follow Section “Support and Require Maven 3.9.11” [91] for upgrade information.

(CMS-27846)

CoreMedia Frontend Workspace

Updated Third Party Dependencies

Third party dependencies of the frontend workspace have been updated to their latest version fulfilling semantic version ranges.

(CMS-29269)

Frontend Workspace now employs the PNPM Catalog Protocol

The Frontend Workspace now employs the usage of the PNPM catalog protocol for all dependencies in package.json of all packages. Using the catalog protocol makes it much easier and less error prone to maintain only one version per dependency in a workspace. For details refer to the original documentation at https://pnpm.io/catalogs

Follow Section “Frontend Workspace now employs the PNPM Catalog Protocol” [91] for upgrade information.

(CMS-29265)

Upgraded eslint to v9

Eslint was upgraded to v9.

Follow Section “Upgraded eslint to v9” [92] for upgrade information.

(CMS-25597)

CoreMedia Deployment

compose-wait command-line tool has been removed from the base image

To make distributed deployments of applications more robust, the java-application-base image contains support to wait for service dependencies. Up to now, this was done using the compose-wait tool. This tool has now been replaced with the much better fit wait4x. To use it the same way like compose-wait, there is an entrypoint script named `wait4x`, that can be included before the application script. See https://github.com/coremedia-contributions/docker.java-application-base for more details.

(CMS-27827)

confd has been removed from the java-application-base image

confd is no longer maintained and mutating files at startup should be avoided. Instead mount static config files if necessary.

(CMS-22475)

Switch studio-client and overview to CoreMedia vendored unprivileged nginx image.

The Docker images for studio-client and overview now use the CoreMedia vendored nginx unprivileged image. Port configuration and other adjustments have been applied to support the unprivileged setup. As a result, these containers no longer use port 80 and have been reconfigured to use port 8080. This change may require adjustments to your existing deployment or firewall settings."

(CMS-16467)

CoreMedia Commerce Hub

Renamed grpc.stubs Properties

The properties prefixed with `grpc.stubs` were renamed to a prefix of `grpc.client`. More specifically, the following properties are affected by this change:

- `grpc.client.default-timeout`
- `grpc.client.service-timeouts`

Rename the properties to conform to their new prefix.

(CMS-27833)

Commerce Hub gRPC Client Requires Proto Reflection V1

Commerce Hub gRPC client was updated to require Proto Reflection V1 on the commerce adapter side. Commerce Adapters using Base Adapter older than v3.0.11 are *not* compatible.

(CMS-27122)

CoreMedia Headless Server

Headless Server: Deprecated content type 'application/graphql+json' not supported anymore

With the necessary update to Spring 7, the already deprecated usage of `MediaType#APPLICATION_GRAPHQL` was removed in Spring 7, thus can't be used anymore. The recommended replacement is `MediaType#APPLICATION_GRAPHQL_RESPONSE`.

Follow Section “Headless Server: Deprecated content type 'application/graphql+json' not supported anymore” [95] for upgrade information.

(CMS-28677)

Headless now support image type transformation for original images (non-crop)

The media delivery of headless formerly offered image type conversation for predefined crops only. This change introduces the ability to transform also the original image.

Follow Section “Headless now support image type transformation for original images (non-crop)” [96] for upgrade information.

(CMS-28460)

Headless Server Content-Schema Deprecations in CM13

Starting with CM13, these deprecated fields were removed GraphQL Type Blob

```
size: Int
```

Please see also corresponding removals at `ContentBlobAdapter` and `ContentMarkupAdapter` in the public Java-API.

GraphQL Type CMMediaImpl

```
captionLegacy: String
captionAsTree: RichTextTree
captionReferencedContent: [Content_!]
```

Follow Section “Headless Server Content-Schema Deprecations in CM13” [96] for upgrade information.

(CMS-28253)

Headless Server Java-API Deprecations in CM13

Starting with CM13, these Java-API deprecations were now applied EvaluationContext (part of rich text transformation)

In class `com.coremedia.caas.richtext.stax.writer.intermediate.eval.EvaluationContext`, the incomplete convenience constructor was now removed. Please use the remaining main constructor instead.

Follow Section “Headless Server Java-API Deprecations in CM13” [97] for upgrade information.

(CMS-27563)

Changed Cache Control Defaults

Cache Control default configuration changed so that the "cache forever" values were increased from a *max-age* of 180 to 365 days and enhanced with 1 hour of *stale-while-revalidate*, 1 days of *stale-if-error*. The blob delivery Cache-Control was also enhanced with the *immutable* directive. The Cache-Control config for Headless Server GET responses was enhanced with 30 seconds of *stale-while-revalidate* and 10 minutes of *stale-if-error*. The Cache-Control config for CAE fragment requests was set to a *max-age* of 2 minutes along with 30 seconds of *stale-while-revalidate* and 10 minutes of *stale-if-error*.

(CMS-27336)

Headless Server: Improved extensibility and flexibility of Blob delivery

With the release of CM13 the Blob delivery has been completely revised. The controller responsible for the Blob delivery is now configured by a number of Spring beans, which are all available as ExtensionPoints. For details please see the new chapter 'Customization of Media Endpoints' Headless Server Manual.

Follow Section “Headless Server: Improved extensibility and flexibility of Blob delivery” [99] for upgrade information.

(CMS-25427)

Miscellaneous

Deprecation of Adaptive Personalization

The integration of Adaptive Personalization is deprecated in favor of the new Native Personalization.

Follow Section “Deprecation of Adaptive Personalization” [99] for upgrade information.

(CMS-29078)

Removed unused code from UriComponentsHelper

Unused code has been removed from the `com.coremedia.objectserver.web.links.UriComponentsHelper`. This results in removed and renamed public methods.

Follow Section “Removed unused code from UriComponentsHelper” [100] for upgrade information.

(CMS-28780)

Adaptive Personalization (p13n) Extensions Disabled By Default

The Adaptive Personalization extensions have been disabled by default in the Blueprint.

Follow Section "Adaptive Personalization (p13n) Extensions Disabled By Default" [100] for upgrade information.

(CMS-28763)

gRPC as an Alternative to CORBA Communication

For internal communication between CoreMedia CMCC components, gRPC has been added as an alternative to CORBA. Communication via CORBA is still available and the default. As long as you plan to stick with CORBA, no changes to your code or operations procedures are required. Should you want to switch to gRPC as your communications protocol between CMCC components, the following sections will provide information and pointer for additional details to do so.

In the course of adding gRPC as a communications protocol, however, breaking changes were introduced:

- The old workflow client API and its implementations have been removed. They were actually unused after removing the *Site Manager*. This API was deprecated with 2406.2.2 and 2506.0.2. This affects the classes in the following packages:
  - `com.coremedia.workflow`
  - `com.coremedia.workflow.common.util`
  - `com.coremedia.workflow.worklist`
- Public workflow server API has been moved. The Java packages of the affected classes didn't change, but they are now contained in Java artifact `com.coremedia.cms:cap-workflow-server` instead of `com.coremedia.cms:cap-workflow`. Consequently, they cannot be used in extensions other than for the workflow server anymore. For clients, please use the Unified API. Documentation for the moved classes used to be at CoreMedia Shared Common API [|https://documentation.coremedia.com/cmcc-12/artifacts/2506.0.0/javadoc/common/com/coremedia/workflow/package-summary.html] and can now be found at CoreMedia Workflow Server API
  .

Follow Section "gRPC as an Alternative to CORBA Communication" [101] for upgrade information.

(CMS-28689)

CORS Configuration Updated

In case you have existing CORS configuration, change your config from `caas.cors.` to **`cors.`** according to the deployment manual.

(CMS-28541)

Removed redundant dependency management from blueprint parent POMs

Imports for Spring dependencies have been removed from the `dependencyManagement` sections from the `blueprint-parent` POMs.

Follow Section "Removed redundant dependency management from blueprint parent POMs" [104] for upgrade information.

(CMS-28515)

Removed util class InputInput

The class `com.coremedia.util.InputInput` has been removed. It's only usage in the content server's `hox.corem.server.media.jdbc.BlobStore`, dynamically determining the length of uploaded blobs, has been replaced with util classes from Apache Commons IO .

(CMS-28497)

Do not start MongoDB and Elastic Worker by default

Due to several changes (Removal of Google Analytics connectors, Disabling of Elastic Social by default, Changing Collaborative Component persistence to SQL) the Blueprint Docker deployment has been adjusted to not start MongoDB and Elastic Worker by default. To start them, please use the compose files `elastic.yml` and `development-elastic.yml`.

(CMS-28438)

Removed deprecated BaseFileSystemConfiguration from ContentHub API

The class `BaseFileSystemConfiguration` has been removed as it is not needed anymore and might cause confusion.

Follow Section "Removed deprecated BaseFileSystemConfiguration from ContentHub API" [105] for upgrade information.

(CMS-28214)

Content hub adapters are no longer delivered as bundled plugins

In previous versions the content hub adapters for youtube and rss were bundled in the studio-server and studio-client. The plugins are no longer bundled and have to be added like other plugins in each project.

(CMS-28213)

CatalogPicture#getUrl() removed from template api

CatalogPicture#getUrl() must not be used from templates, therefore it has been removed from the template api.

(CMS-28177)

Removed Deprecated API: Predicates

The deprecated class `com.coremedia.common.util.Predicates` has been removed from the Java API.

(CMS-28146)

Removal of Analytics Connectors

The integration of Analytics Connectors based on Elastic Core has been removed, i.e. extension `es-alx` and parts of extension `alx`, including the analytics doctypes. Functionality for website tracking is still available and the corresponding documentation has moved to the CoreMedia Blueprint manual.

Follow Section "Removal of Analytics Connectors" [106] for upgrade information.

(CMS-28074)

Elastic Social (es) Extensions Disabled By Default

The Elastic Social extensions have been disabled by default in the Blueprint.

Follow Section "Elastic Social (es) Extensions Disabled By Default" [107] for upgrade information.

(CMS-28072)

Support for MySQL 8.0 and PostgreSQL 13 has been removed

Please update to supported versions MySQL 8.4 / PostgreSQL 16.

(CMS-27936)

The workflowconverter initscript has been removed from the workflow-server image

The recommended way to run the workflowconverter is either by running it in-place at application startup activated by the flag `workflow.server.enable-workflow-converter` or by running the converter tool as an initcontainer or manually, when the workflow-server is stopped.

(CMS-27806)

Confd has been removed from the management-tools image

Confd had been used as a convenience layer to reconfigure the tools based on environment variable based profiles. With confd it was possible configure arbitrary files using a templating mechanism.

Follow Section "Confd has been removed from the management-tools image" [108] for upgrade information.

(CMS-27753)

Middle Tier: Removal of Deprecated API

Follow Section "Middle Tier: Removal of Deprecated API" [108] for upgrade information.

(CMS-27564)

Common Tier: Removal of Deprecated API

The following deprecated classes and methods have been removed:

- `com.coremedia.blueprint.base.navigation.context.ContextStrategy#selectContext`
- `com.coremedia.cache.CacheCapacityConfigurer`
- `com.coremedia.cap.common.Blob#getSize`
- `com.coremedia.cms.common.plugins.beans_for_plugins.CommonBeansForPluginsConfiguration`
- `com.coremedia.common.logging.MarkerFactory#getMarker`
- `com.coremedia.dispatch.InterfaceOnlyPredicate`
- `com.coremedia.livecontext.ecommerce.asset.AssetService#getDefaultPicture`
- `com.coremedia.livecontext.ecommerce.common.CommerceIdProvider#formatProductTechId`
- `com.coremedia.livecontext.ecommerce.common.CommerceIdProvider#formatProductVariantTechId`
- `com.coremedia.livecontext.ecommerce.common.CommerceIdProvider#formatCategoryTechId`
- `com.coremedia.livecontext.ecommerce.user.UserContext#getCookieHeader`
- `com.coremedia.livecontext.ecommerce.user.UserContext.Builder#withCookieHeader`
- `com.coremedia.xml.UnsafeJdkJaxpFactory`
- `com.coremedia.xml.XmlUtil5#escape`

Follow Section "Common Tier: Removal of Deprecated API" [112] for upgrade information.

(CMS-27557)

Third-Party Update: Apache Solr 9.10.0

Apache Solr has been updated to version 9.10.0 and the coremedia/solr-base Docker image with tag 2.6-cm-9.10.0 has been released. As part of this change, the following transitive dependencies of Apache SolrJ have been updated:

- Apache Zookeeper 3.9.4
- Eclipse Jetty 10.0.26

Follow Section "Third-Party Update: Apache Solr 9.10.0" [112] for upgrade information.

(CMS-27540)

Discontinue base image flavors

The java-application-base image no longer provides jdk based image flavors. The jre for corretto and temurin is an adjusted custom jre based on the default jre module set with the addition of jshell.

(CMS-27499)

Switched to spring-grpc

Switched from grpc-spring to spring-grpc. This change causes that properties below `grpc.server` and `grpc.client` are no longer relevant. Instead the properties below ``spring.grpc.server}}and {{spring.grpc.client}}must be used respectively. Along with this change, the commerce hub client setting {{com.coremedia.blueprint.base.livecontext.client.settings.CommerceSettings#getEndpoint}}was removed. Use `{{com.coremedia.blueprint.base.livecontext.client.settings.CommerceSettings#getEndpointName}}instead and configure an ops-friendly symbolic name, which is then configured as a named gRPC channel below {{spring.grpc.client.channels``. More specifically, this means that your ``commerce}}struct’s {{endpoint}}value is no longer picked up to establish a commerce connection. Introduce a String property named {{endpointName}}in your {{commerce}}struct and give it the value of a gRPC channel, such as {{hybris}}or {{sfcc``.

(CMS-27495)

Java 21 Support

Starting with CMCC v13.2512.0.0 the required JDK to build the workspace and run the applications is Java 21 LTS Release.

Follow Section “Java 21 Support” [114] for upgrade information.

(CMS-27319)

Removal of deprecated Commerce Hub Data properties

The following deprecated properties have been removed. Use their replacements in parentheses '(…)' instead.

- `commerce.hub.data.deadline` (`grpc.client.default-timeout`)
- `commerce.hub.data.services-without-timeout` (`grpc.client.service-timeouts`)

The property `commerce.hub.data.endpoints` has been removed without direct property replacement. Instead, use the `endpointName` of the `commerce` settings struct (respectively `com.coremedia.blueprint.base.livecontext.client.settings.CommerceSettings#getEndpointName`) to configure the Commerce Hub connection.

(CMS-27298)

Removed Deprecated API for Custom Proactive Engine Application

Custom Proactive Engine / Persistent Cache applications are not supported anymore. Proactive Engine and Persistent Cache are internal technical components of the *CAE Feeder* that cannot be used in custom applications outside of the *CAE Feeder* anymore. The Maven artifact `com.coremedia.cms:proactiveengine-component` and deprecated public Java API have been removed.

Follow Section “Removed Deprecated API for Custom Proactive Engine Application” [115] for upgrade information.

(CMS-27251)

Studio-Client Docker build improvements

The Studio-Client Docker build has been simplified to provide a better convenience when building the image. All three Dockerfiles (Dockerfile, Dockerfile.tooling and Dockerfile.tasks) have been merged into a single Dockerfile which is then preconfigured using a Docker Bakefile.

Follow Section “Studio-Client Docker build improvements” [117] for upgrade information.

(CMS-26030)

Support for Microsoft SQL Server (MSSQL) has been removed

As announced, the support for the database Microsoft SQL Server (MSSQL) has been removed from the product.

(CMS-25905)

Change Storage Layer of Collaborative Components to SQL

The Collaborative Components (My Edited Content, Projects, Workflow Archive, Notifications) now store their data in an SQL database instead of MongoDB. MongoDB and the In-Memory replacement are not supported any longer for these features, but can still be enabled for instance to prepare a data migration. For instructions on data migration and further details please see the section *Collaborative Components* in the *Operation Basics* Manual.

(CMS-25582)

4.1.2 Changes and Improvements

CoreMedia Core

JShell Enhancements

Aligned with the recent enhancements for CMCC v12.2506.0.0 of configuration options (for example, via environment properties but also better support for properties stored in `.cm.jshell.properties`), the API of the `coremedia` namespace has been enhanced by two new connect methods:

- `connect(username, password):CapConnection`
- `connect(Map<String, String> params):CapConnection`

The former makes the `repository.url` obsolete, which will then be determined from the outer configuration (like from environment variables or from `.cm.jshell.properties`). The latter leverages the map-like configuration, that already exists for `Cap.connect(...)` to be also available in JShell. As always, the benefit of using `coremedia.connect(...)` over `Cap.connect(...)` is not only automatic close of the connection when leaving JShell, but also heaving a fully-fledged application context at hand, like accessing the `SitesService` via `coremedia.sites` (or short: `cm.sites`).

(CMS-29030)

Third-Party Update: Apache Tika 3.2.3

Apache Tika has been updated to version 3.2.3. As part of this change, transitive dependencies of Apache Tika have been updated.

Updated dependencies:

- Apache Tika 3.2.2
- jsoup 1.21.2

If you use these libraries in project code, please check their respective release notes for changes and upgrade information.

(CMS-28582)

Deprecated Content Feeder Configuration Properties

The *Content Feeder* configuration properties `feeder.content.issues.reindex-time-max-percentage` and `feeder.content.issues.reindex-time-window` have been deprecated. They have been replaced by properties `feeder.content.background-feed.ISSUES.max-time-percentage` and `feeder.content.background-feeder-max-time-window`, which should be used instead.

(CMS-28465)

Third-Party Update: Apache Tika 3.2.2 and Transitive Dependencies

Apache Tika has been updated to version 3.2.2. As part of this change, transitive dependencies of Apache Tika have been updated.

Updated dependencies:

- Apache Commons CSV 1.14.1
- Apache Commons IO 2.20.0
- Apache Commons Lang3 3.18.0
- Apache Mime4J 0.8.13
- Apache PDFBox 3.0.5
- Apache Tika 3.2.2
- ASM 9.8
- jsoup 1.21.1

If you use these libraries in project code, please check their respective release notes for changes and upgrade information.

(CMS-28431)

Plugins can Contribute Document Type Aspects

There is a new Extension Point for Plugins. The `content-server` Application Plugin of a Plugin Bundle can now contain resources matching the classpath pattern `classpath*:framework/doctypes/*/.xml`. All these files will be picked up as document type definitions, which can contain new document types as well as document type aspects (see Content Server Manual / 4.2.3 Attaching Properties to Existing Content Types). The extended document type model becomes active on first Content Server start after the Plugin has been added.

Important note: If a Plugin comes with custom document type definitions, once you have activated it and started the Content Server, its database is adjusted accordingly. You must not disable the Plugin afterwards, or Content Server might refuse to start, because the additional document type definitions are missing. In case you want to remove such a Plugin, consult the Content Server Manual about adjusting the database after document type changes.

(CMS-27872)

Updated Project Dependencies

Updated several dependency and plugin versions:

- Angus Activation → 2.0.3
- Angus Mail → 2.0.5
- Apache HttpComponents Client 5 → 5.6
- AspectJ → 1.9.25
- AssertJ → 3.27.6
- Awaitility → 4.3.0
- Bouncy Castle Java → 1.83
- Byte Buddy → 1.17.6
- Caffeine → 3.2.0
- ClassMate → 1.7.3
- Commons CLI → 1.11.0
- Commons Codec → 1.20.0
- Commons Collections → 4.5.0
- Commons Compress → 1.28.0
- Commons Exec → 1.6.0
- Commons IO → 2.21.0
- Commons Lang → 3.20.0
- Commons Pool → 2.13.1
- Commons Text → 1.15.0
- CoreMedia Adapter for HCL Commerce → 3.0.14
- CoreMedia Adapter for Salesforce Commerce Cloud B2C → 3.0.16
- CoreMedia Adapter for SAP Commerce Cloud → 3.0.14
- CoreMedia Client API and Mock Adapter for Commerce Hub → 3.0.16
- Enterprise Security API → 2.7.0.0
- Error Prone → 2.45.0
- Extended Scalars for GraphQL Java -> 24.0
- FreeMarker → 2.3.34
- Google Analytics Data Client for Java → 0.75.0
- Google API Common for Java → 2.50.2
- Google API Extensions for Java → 2.67.2
- Google Auth Library for Java → 1.37.0
- Google Common Protos Client for Java → 2.63.2

- Google HTTP Client Library for Java → 1.47.1
- Google OAuth Client Library for Java → 1.39.0
- GraphQL Java → 25.0
- GraphQL Java DataLoader → 5.0.3
- gRPC-Java → 1.78.0
- Gson → 2.13.2
- Guava → 33.5.0-jre
- Hibernate ORM → 7.1.11.Final
- Hibernate Validator → 9.1.0.Final
- Jackson → 2.20.1
- Jakarta Servlet API → 6.1.0
- Jakarta Validation API → 3.1.1
- Javassist → 3.30.2-GA
- JBoss Logging → 3.6.1.Final
- Jib Maven Plugin → 3.5.1
- JLine → 3.30.6
- JSON-Java → 20251224
- JUnit Jupiter → 6.0.1
- Logback → 1.5.22
- Micrometer → 1.16.1
- Mockito → 5.21.0
- Netty → 4.2.9.Final
- Plugin Framework for Java → 3.14.0
- Protobuf Maven Plugin → 4.1.2
- Protocol Buffers for Java → 4.33.2
- Reactor Project BOM → 2025.0.1
- Resilience4j → 2.3.0
- SLF4J → 2.0.17
- SnakeYAML → 2.5
- Spring AI → 2.0.0-M1
- Spring Boot → 4.0.0
- Spring Data BOM → 2025.1.1
- Spring for GraphQL → 2.0.1
- Spring Framework → 7.0.1
- Spring gRPC → 1.0.0
- Spring Security → 7.0.0
- Spring Webflow → 4.0.0
- Tomcat → 11.0.15
- Vavr → 0.11.0

(CMS-27700)

Session Check for Blob Download from Content Server Can Be Disabled

Blob Downloads from Content Servers are verified for a valid login session on each call. With the new configuration option `cap.server.blob-download-session-validation`, this session validation can be disabled. The default for the option is `true`, i.e., each download request is verified for a valid login session. If set to `false`, the session validation for downloads is disabled. This is useful on *Replication Live Servers* to allow for load-balanced blob download. It must only be set to `false` if corresponding servers are appropriately secured by firewalls, private networks, or the like. Otherwise, public unsecured read access to blobs could be possible. *Blob Uploads* are *always* secured by session validation, regardless of the option's setting.

(CMS-27698)

MongoDB update to version 8.0

MongoDB has been updated to version 8.0 for the Docker Deployment. Note that an upgrade is only possible from a 7.0.x version. An upgrade from an earlier version needs to be done successively. Please refer to the official MongoDB documentation on how to upgrade: https://www.mongodb.com/docs/manual/release-notes/8.0-upgrade-standalone/ The compatibility changes are described here: https://www.mongodb.com/docs/manual/release-notes/8.0-compatibility/

(CMS-25999)

Add Checkbox To Publication Workflow Window To Include Referenced Updated Content

Previously, only unpublished referenced content items were included in the publication. When the new checkbox is enabled, also referenced content items are included that have been updated since their last publication.

The depth for following such references is limited by Studio Setting `publication.maxUpdatedLinksDepthToCompleteChangeSet` or Spring property `studio.rest.changeset-max-updated-links-recursion-depth`.

For more details see the section “Configuring Workflow Validation” / “Publication Settings” in the Studio User Manual.

(CMS-17555)

CoreMedia Content Application Engine

Fixed SSRF vulnaribility in CodeResourceHandler

Static files other than below the CAE root could be retrieved by manipulating the passed path.

(CMS-28554)

CoreMedia Studio Client

Coremedia npm repository changed

The npm repository switched to repository.coremedia.com/nexus/repository/coremedia-npm (see documentation).

The former repository npm.coremedia.io is also still available but might be removed in the future.

(CMS-29266)

Removed Outdated Documentation about Teaser Text Delegation

Outdated comments for the properties “delegatePropertyName” and “delegateExpression” have been removed from the "TeaserOverlayContainer.ts".

(CMS-28961)

Pendo plugin no longer bundled by default

The Pendo-based tracking functionality and in-app guides are no longer shipped as a bundled Blueprint plugin and are no longer installed by default during deployment. From **CMCC 13** onwards, the Pendo plugin is deprecated.

Note: We are evaluating alternative solutions by utilizing our in-house AI assistant “KIO”.

(CMS-28960)

Increase Tool Tip Delay

The delay for showing tooltips has been released to 1500ms.

(CMS-28655)

Improved Shortcuts Description

Improved shortcut description "Show opened tabs menu".

(CMS-28416)

New icon design

Improving User Experience through new icon design.

(CMS-28088)

Character Counter for TextFields

The components for textfields and textareas (`StatefulTextField` and `StatefulTextArea`) may now show the length of the input text if a maximum length for the field is set. This feature has to be enabled for each field manually by setting the `showCharacterCounter` property. The maximum length can be set via the `characterLimit` property.

You can also set those properties on any `StringPropertyField` directly. This way, the `Length` attribute of the corresponding `StringProperty`, set in `blueprint-doctypes.xml`, will automatically be taken into account to define a default for the `characterLimit` property. E.g. The article title field will automatically have a maximum length of 512 characters.

Please note, that the character counter will only show if the field is focused.

(CMS-27970)

CoreMedia Studio Server

Fixed Single-Selection Mode for the Taxonomy Chooser

The single-selection mode that is available for the taxonomy chooser component has been fixed. In addition to that, the *singleSelection* configuration option is autmatically derived from the *maxCardinality* flag of the taxonomy link list. So when the *maxCardinality* is set to *1* for the *TaxonomyLinkListPropertyField*, the dialog switches to single-selection too.

(CMS-22696)

CoreMedia Workspace

For kubernetes test system deployments a new overview app will be installed

The new overview application renders sites, administration and topology information dynamically retrieved from the Kubernetes API server.

(CMS-28322)

Cleaned up dependency-management

Duplicated dependency management in `blueprint-parent` poms for `spring-boot-starter-tomcat` and `tomcat-embed-*` artifacts has been removed. These dependencies are already managed in `common-thrird party-bom`.

(CMS-25927)

CoreMedia Commerce Hub

Dedicated Cache Class for Commerce Objects

Most of the commerce entities come along with their own cache class that can be used for capacity and cache duration configuration, which still should be used.

Others couldn't be configured explicitly and re-used the default configuration for "java.lang.Object". Now they can be configured via "ecommerce.object". The default configuration

```
cache.capacities.ecommerce.object=10000
```

should be sufficient for most cases.

No action is required.

List of cache keys that are affected:

```
com.coremedia.blueprint.base.ecommerce.catalog.CmsCatalogService.CommerceBeanCacheKey
com.coremedia.blueprint.base.ecommerce.catalog.RootCategoryContentCacheKey
com.coremedia.blueprint.base.livecontext.client.data.grpc.DataEndpointCacheKey
com.coremedia.blueprint.base.livecontext.ecommerce.common.CommerceSettingsCatalogConfigListCacheKey
com.coremedia.blueprint.base.livecontext.ecommerce.common.SiteToCommerceConnectionCacheKey
com.coremedia.blueprint.base.livecontext.ecommerce.common.SiteToStoreContextCacheKey

com.coremedia.blueprint.base.livecontext.ecommerce.common.StoreIdAndLocaleToSiteCacheKey

com.coremedia.livecontext.context.LiveContextResolveContextStrategy.CommerceContextProviderCacheKey
com.coremedia.livecontext.view.ConfiguredContentTypesCacheKey
```

Upgrade Information

(CMS-28178)

CoreMedia Headless Server

Faceted search query returns facet content

A faceted search query returns the facet content, if applicable.

(CMS-28152)

Taxonomy localization

Taxonomy localizations can now be retrieved directly on the graphql type, via field `CMTaxonomy#localization(locale:String!)` or via field `CMTaxonomy#localizations`, which provides all available localizations. The method `TaxonomyAdapter#getLocalizedTaxonomy` has been deprecated in favor of `TaxonomyAdapter#getLocalized`, which returns a specific type `LocalizedTaxonomy` instead of a generic Map.

(CMS-28142)

Miscellaneous

Fix default for KIO Playbook Github Token

Fixed default for KIO Playbook GitHub Token in blueprint example deployment file.

```
KIO_BACKEND_PLAYBOOKS_GITHUB_TOKEN: ${KIO_GITHUB_TOKEN:-false}
```

If not configured, the auto config relying on this property is skipped as expected.

(CMS-28987)

Resilience4j Update

Resilience4j has been updated from version 1.7.0 to 2.3.0 for the studio-server and the content-server.

(CMS-28786)

Update of JUnit Jupiter

JUnit Jupiter has been updated to new major version 6.0.

Custom code that involves usage of changed JUnit Jupiter API may need to be adapted, see https://docs.junit.org/current/release-notes/index.html.

(CMS-28721)

Updated MySQL Driver

Updated MySQL driver to `com.mysql:mysql-connector-j:9.4.0`

(CMS-28559)

Configurable Cache Duration for Outdated Transformed Blobs

Added a new configuration property `delivery.outdated-transformed-blob-cache-duration` to control how long outdated or yet unknown transformed blobs are to be cached in a multi-CAE scenario (`cae.single-node=false`). Before this fix, CAE send a cache-control header with `max-age` of 0 seconds. This value is now configurable. Consider that a yet unknown cache should not be cached for long because the CDN should ideally cache the correct transformed blob for a given URL. The cases to cover are that (1) either a more up-to-date CAE generated the link and the CAE answering the request for the transformed blob does not yet know about the updated blob and (2) a client has stored a very old version of a transformed blob URL for which the blob data is no longer cached by the CDN. A CAE cannot distinguish between these two cases but serves the currently (either outdated or newer) known transformed blob. The new configuration property allows projects to optimize for scenario (1) or (2) by setting a (1) short duration or (2) a long duration.

Configuration example:

```
# Set cache duration to 1 hour
delivery.outdated-transformed-blob-cache-duration=1h
```

Upgrade Information

(CMS-28032)

JUnit Jupiter: Updated to 5.13.2

JUnit Jupiter has been updated to 5.13.2. Note, that this release introduces deprecations, for example, for `ArgumentsProvider` so that you should implement another signature of `provideArguments`.

For details, see the corresponding release notes: JUnit 5 Release Notes.

(CMS-28006)

Updated Hibernate to 7.1

`jakarta.persistence:jakarta.persistence-api` has been updated to `3.2.0`

`org.hibernate.orm:hibernate-core` has been updated to `7.1.4.Final`

`hypersistence-utils` have been updated to `io.hypersistence:hypersistence-utils-hibernate-71:3.11.0`

(CMS-27967)

Third-Party Update: commons-io 2.19.0

Apache Commons IO has been updated to 2.19.0.

(CMS-27839)

New methods Content#getLatestVersion, Content#getLatestOrWorkingVersion, Content#getCheckedInOrWorkingVersion and Content#getCheckedOutOrWorkingVersion

The `com.coremedia.cap.content.Content` interface has been extended with methods `Content#getLatestVersion`, `Content#getLatestOrWorkingVersion`, `Content#getCheckedInOrWorkingVersion` and `Content#getCheckedOutOrWorkingVersion`. For details see javadoc.

(CMS-27810)

Removed unused legacy CORS configuration properties

The deprecated and unused `cae.cors.` **(CAE) and `caas.cors.`** (Headless Server) configuration properties have been removed.

Since CMCC-12 release 2506.0.0, these delivery-specific CORS configuration properties were unused as they were replaced by the generalized `cors.*` configuration properties.

Please see the CORS Properties reference in the Deployment Manual.

(CMS-27692)

Third-Party Update: Checker Framework

The Checker Framework has been updated to version 3.42.0.

(CMS-27505)

Third-Party Update: Eclipse Glassfish CORBA ORB

The third-party library org.glassfish.corba has been updated to version 5.0.0

(CMS-27460)

The git-commit-id-plugin has been removed from the maven build system

The git-commit-id-plugin has been removed from the maven build. If you want to add git information to the build, it should be set on an application basis and not with a global state, otherwise every commit will change the image digest and every commit will require a redeployment of every image.

In order to get this information at runtime, we recommend to use

```
git log -n 1 --pretty=format:%H -- <folder>
```

and pass it to the jib-maven-plugin to set it as an environment variable and a label.

```
<configuration>
  <container>
    <environment>
      <INFO_COMMIT_ID>${git.commit.id}</INFO_COMMIT_ID>
    <environment>
    <labels>
      <INFO_COMMIT_ID>${git.commit.id}</INFO_COMMIT_ID>
    </labels>
  </container>
</configuration>
```

There is a caveat of this approach is that there are shared modules which might change and then would not reflect in this property. Alternatives are OCI manifest annotations which can be added after the build without changing the images digest but are not possible to access at runtime.

(CMS-27281)

jshell has been added to all java-application-base image flavors

Both the corretto-jre and the temurin-jre base image flavors now contain jshell.

(CMS-27258)

a new maven profile to build multi-arch images has been added

The new profile `multi-arch` complements the `default-image` profile and works only together with the `build` goal. If activated, the `application.image-arch` property has no effect and instead each image will be build for `linux/amd64` and `linux/arm64` architectures

(CMS-25935)

The cm based command-line tools have been moved to the corresponding server application images

To provide a better Kubernetes integration, the tools have been moved to the corresponding server images. For backwards compatibility, the `management-tools` image will still be provided.

(CMS-25844)

LdapUserProvider Connection Improvements

LDAP roundtrips ere executed with a timeout now, in order to avoid blocked threads in case of certain problems. The timeout (`ldap.timeout`) as well as the maximum number of connections (`ldap.max-connections`) is configurable now. However, the default values are supposed to be suitable for most projects, and there should be no need to change them unless concrete problems occur. There is another new configuration property `ldap.referral-throw-handling` to control the details of referral handling. It is still experimental, because the details concerning referrals depend on the particular setup of the

LDAP servers, and this has not yet been tested exhaustively. But if you encounter problems with the default behaviour, you should give it a try.

See `com.coremedia.ldap.LdapUserProviderConfigurationProperties` for details.

(CMS-23359)

Running with 2 retries at fixed delays of 1 and 5 seconds

(CMS-22581)

4.1.3 Fixed Issues

CoreMedia Core

Workflow Action Exception Logging

Fixed a bug in the logging of some workflow action exceptions where the underlying cause was not visible.

(CMS-28851)

XLIFF-Import: Relaxed xlink:href and xlink:role Validation

When importing XLIFF that includes a CoreMedia Rich Text 1.0 property value having an anchor with `href` like `href="https://"`, you may have observed an error such as:

```
[Fatal Error] cvc-datatype-valid.1.2.1: 'https://' is not a valid value for
 'anyURI'.
```

The cause is a mismatch between the CoreMedia Rich Text 1.0 specification (given as DTD) and the XLink namespace, thus, CoreMedia Rich Text 1.0 allows any string value as `xlink:href` while the XLink schema-based namespace does not.

As fix to grant that “any exported CoreMedia Rich Text 1.0” can be imported again as XLIFF, we relaxed the schema validation for `xlink:href` and also for a comparable mismatch in `xlink:role`.

No adaptations to existing XLIFF files are required.

(CMS-28696)

Deprecated Remote Client Actions

Remote client actions for workflows have been deprecated. There are no clients implementing this API anymore except the workflow-server itself.

To this end the `AbstractClientAction` and all derived classes have been deprecated for removal, as they come with some unnecessary overhead like timeouts, potentially causing issues during execution.

The `ApproveResources` and `PublishResources` Actions have been re-implemented as `Approve` and `Publish`, based on `LongActionBase`. If you implemented actions based on `AbstractClientAction` yourself, please also consider migrating them to extend `LongActionBase` instead.

The built-in workflows `studio-simple-publication.xml` and `studio-two-step-publication.xml` have been adjusted to use the new Actions, so you should re-upload these workflows. Please consider updating usage of these actions in your own custom workflows in the same way.

(CMS-27908)

Increased Maximum Content-Type Name Lengths

Allowed maximum content-type name lengths got increased from 18 to 56 for all supported databases.

(CMS-27798)

Extended Output of cm dump Tool for Workflow Processes

The `cm dump` and the `cm processes --verbose` command-line tools were extended to output more data about workflow processes. The output now includes “task states”, “task escalation exceptions”, and “task warnings”. This can be helpful to analyze the current state of a running workflow, including errors that happened during the execution of a workflow task.

Previously, such information was only written if these tools were used to dump the workflow tasks itself, for example if `cm dump` was called with a task id, `cm dump --verbose` was used to dump a process, or with `cm processes --very-verbose`.

(CMS-25759)

Fixed Bug In Integrated Workflow Conversion With PostgreSQL

Integrated workflow conversion (enabled in the workflow-server with `workflow.server.enable-workflow-converter=true`) has been fixed for PostgreSQL.

(CMS-25566)

Archive Broken Workflow Definitions

Parameter `-A` has been added to the `workflowconverter` which can be used to potentially keep broken workflow definitions for usage in the workflow archive, so that already archived workflows of a broken workflow definition can still be accessed.

For details, please see section “Workflowconverter” in the Workflow Manual.

(CMS-23988)

Optionally Pass Authentication Data for Content Server's HTTP Endpoints in Request Header Fields

Each *Content Server* provides HTTP endpoints for blob up- and download as well as for processor usage data. These are internally used endpoints that are not supported for use by custom clients. You may configure whether authentication data for those endpoints is to be sent as part of the URL (which was the only way prior to CMCC 13) or as a request header field. While sending the token as a query parameter is inherently insecure, it is the default to keep backward compatibility. Unless connection to an older server (prior to CMCC 13) is required, it is recommended to set these properties to *false*.

See *Deployment Manual*, section *4.4.7 Securing Session Authentication for Content Server HTTP Endpoints* for details of configuration.

(CMS-16480)

CoreMedia Content Application Engine

Configured Forced Response Character Encoding for CAE

To prevent SRI mismatches due to incorrect character encoding, the CAE is now configured to force character encoding by default. Also see https://docs.spring.io/spring-boot/appendix/application-properties/index.html#application-properties.web.spring.servlet.encoding.force

(CMS-29178)

Changed Themes Target Folder Name

In the CAE local development workflow with local resources an issue has been observed where static resources were not found. Therefore, the folder `frontend/target/resources/themes` has been changed to `frontend/target/resources/Themes`. It may be necessary to delete the target folder of the frontend workspace once after upgrade.

(CMS-28937)

Correction of CAE and HLS Cache Header max-age

Cache Header *max-age* of resources with validity date delivered by CAE and HLS could be off by up to one second, invalidating a resource in caches too soon, i.e., before an updated version of the resource becomes available. This has been fixed by adding a second to the *max-age*.

(CMS-28897)

Fixed Parameter Resolution of Dynamic Includes

When using dynamic includes on the Preview CAE the parameters (required to serve personalized content) were lost. This bug has been fixed so that parameter resolution for dynamic includes functions properly on Preview CAE.

(CMS-28288)

Fixed JMX issue with CAE startup

In rare cases the CAE didn't start up the application context due to an issue with anonymous LinkFormatter instantiation in JMX configuration.

(CMS-28160)

Default configuration of cache-control headers fixed

In Live CAE the default configuration of cache-control headers for css and javascript URLs contained an incorrect pattern. This led to exceptions in the log and the values could not be read.

(CMS-27938)

Removed duplicate of `userVariant`

The query parameter `userVariant` was added twice in a redirect (e.g. the preview call in studio preview). This has been cleaned up and the `userVariant` is now only added once.

(CMS-16789)

CoreMedia Studio Client

Improved Shortcuts Description

Dialogs can be moved via shortcuts. The description for these shortcuts have been improved in the the shortcut descriptions overview.

(CMS-29040)

Improved Dialog Accessibility

The close button of all Studio dialogs has an aria label now and can be accessed via keyboard navigation.

(CMS-29037)

Fixed an issue that time zone names were not propertly localized in studio-client

The issue has been fixed by modifying `apps/studio-client/shared/js/timezones-plugin/src/registerTimeZones.ts`. In case you have copied the implementation, please update these changes accoridngly.

(CMS-28912)

Fixed Workflow Sidebar Footer

Fixed issue that the workflow sidebar footer toolbar was not displayed for users, that do not read permissions for a workflow.

(CMS-28699)

Fixed Annotated LinkList Textfields

Fixed issue that the text of an INPUT field within an annotated link list was not selectable via mouse.

This issue has been fixed for Firefox now too.

(CMS-28417)

Fixes: Link Balloon closes on content with multiple richtext editors when opening the library/control room

When the Richtext-Editor, which has a CKEditor5RichTextPropertyFieldDelegatePlugin, looses focus, then the delegated value is applied. Which causes the Link Balloon to close. If the value would not change, avoid applying the value.

(CMS-28369)

Fixed Miscellaneous Aria Roles and Labels

- Improved the aria label for the type icon column by adding role “img” and an aria label.
- Added aria labels for the image editors rotation actions.
- Fixed preview options toolbar aria label.
- Fixed toolbar overflow button label.
- Changed the header toolbar aria label from **Kopf** to **Site Auswahl, Guides, Info-Center und Benutzermenü**.
- Added aria role “button” and aria label for the close button of work area tabs.

(CMS-28259)

Repository Tree Accessibility

The aria role of repository tree items has been changed to “treeitem”.

(CMS-28258)

Fixed Widget Button Tooltips

The “Edit” button for dashboard widget has been fixed so that the tooltip adepts depending on its mode. This includes the aria label too.

(CMS-28257)

Fixed Miscellaneous Aria Roles and Labels

- Added aria label for the user menu.
- Added aria labels for the user menu items.
- Added aria labels for direct input fields (everywhere, e.g. also projects and struct editor).
- Added aria labels for column headers, telling the users what the name of the column is they are sorting.
- Fixed aria label for the collapsible filter panels. The titles have been set properly so that the aria label of the expand/collapse button contains the name of the panel now.

(CMS-28254)

Fixed HTML "lang" Attribute

The “lang” attribute is now set to the “html” element instead of the “body” element where it belongs according to w3c standards.

(CMS-28252)

Tooltips are now activated on Focus

For better accessibility, the tooltips are now not only shown on a mouse hover but also on keyboard focus.

(CMS-28250)

Welcome Widget is now accessible

The links in the welcome widget on Dashboard can be navigated to via keyboard and activated pressing the enter button.

(CMS-28243)

Fixed Library Column Accessibility

The accessibility for columns inside the library (and other grid components) has been fixed: If a column is sortable, the sort order can be toggled via key navigation (Enter key).

(CMS-28239)

Fixed Library Context Menu Accessibility

The accessibility for the context menu in the library has been fixed. The context menu can now be opened via the keyboard shortcut CTRL+Enter.

(CMS-28238)

Added Missing QuickCreate Parameters

Added missing parameters from the *QuickCreateAction* to the the *QuickCreateLinkListMenu* so that all configuration possibilities are available there too.

(CMS-28133)

Image Editor: wrongly sorted image transformation strings

The image editor calculates transformation strings if they do not already exist as soon as a user checks out a picture document. The new transformation strings had a wrong order which is now fixed. The order calculated by studio-client is now the same as calculated by studio-server.

(CMS-27993)

Skip unnecessary save action

Method setValue of CKEditor5RichTextArea does not trigger a save action anymore.

(CMS-27643)

Fixed: CKEditor drag&drop imports contents from other systems (e.g. Content-Hub, Commerce) to the site root folder.

When drag&drop from e.g. Content-Hub, Commerce to a CKEditor contents were imported to the site root folder. Now the contents are imported to the same folder as the edited content item.

(CMS-27485)

BoundRadioGroup DefaultValue Warning

The component **BoundRadioGroup** is logging a warning now, in case no default value has been set. If this value is not set, the change event handling might not work properly and does not result into content updates.

(CMS-27053)

Fixed Miscellaneous Struct Editor Issues

- Fixed issue that the markup editor for did not show values or did show values for other markup fields or did not save these.
- Fixed errors that happened for markup lists.
- Fixed the error that when a property editor dialog was opened for a struct property (e.g. markup or date fields) and the node was cut or deleted, the dialog remained open.

(CMS-26035)

Workflows in ControlRoom inbox are now correctly displayed

When several workflow are created at the same time (e.g., a translation is triggered against multiple target languages), then there are several workflows with the same creation date. When completing one of them it could remain in the inbox even after it is completed. Now the malbehavior is fixed.

(CMS-25975)

Fixed Tree Lazy Loading

Fixed lazy loading issues of the repository tree.

(CMS-25609)

Fixed Connection Recovering

The connection error handling of the Studio client has been improved so that erroneous requests that failed because of connection issues can recover.

(CMS-24059)

Fixed: postMessage origin error

Analysis: InnerPreviewPanelBase#updateNativeScrolling() attempts postMessages too early on the preview iframe. At that point, the iframe is not yet set up and cannot property determine its origin. Fix: Added an additional check for the current origin of the IFrame.

(CMS-21564)

CoreMedia Studio Server

Users can no longer delete their own Home folder in Studio.

An issue has been fixed where users could delete their own Home folder in Studio. This could lead to unexpected Studio behavior and loss of user-specific settings. Studio now prevents deletion of the Home folder.

(CMS-29318)

Replacing workflow start validators via Customize-Annotation leads to startup errors of Studio-Server

Fixed a bug, where replacing a workflow validator via Customize-Annotation failed because the replacing validator bean could not be found by the original name of the replaced bean.

Updated documentation on how replacement via customizer works and its side effects. See *'Content Application Developer Manual'* section *'Customizer'*

(CMS-28777)

CKEditor5: Link-Balloon layout issue has been fixed

A layout issue where the content-link button would take too much space and move the "Insert/Update" button out of the balloon (especially with German language) has been fixed.

(CMS-28685)

Fixed Search by Path in Studio Folder Chooser

Fixed a bug that made it impossible to search by path for folders in the folder chooser, for example to find the target folder in the create new document dialog.

(CMS-28336)

Custom content creation paths from Editor Preferences fixed

An issue has been fixed where custom contentCreationPaths configured via Editor Preferences were not applied when creating new content in Studio. With this fix, custom default folders defined in Editor Preferences work as expected again. Global and site-dependent configurations were not affected.

(CMS-27829)

Adjusted storage of editorial-comments timestamps to prevent issues with daylight saving times

The Hibernate time zone for editorial-comments has been set to UTC by setting the property `editorial-comments.jpa.properties.hibernate.jdbc.time_zone=UTC`.

Previously, the time zone was depending on the time zone of the JVM.

Externally visible timestamps do not change in the application. Existing timestamps in Oracle have to be adjusted after the time zone change, which is done automatically on start of studio-server.

(CMS-27609)

Removed obsolete configuration from Studio Server

Removed obsolete config `solr.cae.collection` from Studio Server.

(CMS-25650)

Fixed Order of Uploaded JSON Settings

When uploading a JSON file to Studio the order of the entries is preserved now.

(CMS-25282)

CoreMedia Headless Server

Changed CaasConfig#remoteLinkDataLoader

The bean creation of `CaasConfig#remoteLinkDataLoader` has been changed to comply with the API of `DataLoader`, this includes a change of the return type from `DataLoader<String, Try<String>>` to `DataLoader<String, String>`.

(CMS-28787)

Headless responds to HTTP-Head over REST now as in CM11

In CMCC 11 (2310), Headless Server responded to HEAD requests over the REST-API with 200 and 404 as long as JSLT was enabled (which is the default).

With the introduction of Spring-GraphQL, the HTTP transport layer was redefined by Spring. As a consequence, its basic implementation was only able to respond to POST requests. The ability to respond to GET requests was add by CoreMedia but support for HEAD requests was omitted, as such requests to a GraphQL endpoint are neither covered by the GraphQL-over-HTTP spec, nor does it make sense to respond to a HEAD request on a plain GraphQL endpoint.

Nevertheless, supporting HEAD requests makes sense for the REST-API and in CMCC 11 (2310), Headless Server responded with 200 or 404 to a HEAD request as long as JSLT was enabled. Note that the response was always 200, when JSLT was disabled.

This changed behavior was now reestablished for CMCC 12.

Also note that HEAD requests to the REST-API are not recommended and you should strive for solutions avoiding HEAD requests!

Upgrade Information

(CMS-28337)

Richtext headings are resolved with inline handlers

Before, links to content in richtext headlines were not resolved via GraphQL in fields `text`, `textReferencedContent` and `textAsTree`. The content links are resolved correctly now together with other inline elements, by adding the inline handlers to the headline context.

(CMS-28203)

Miscellaneous

Extended document types for XmlUapiConfig

The default test content type model that is provided by `XmlUapiConfig` has been extended with a `base` link property in the `SimpleSiteContent` content type. This property supports basic tests involving content variants.

(CMS-29384)

Fixed locale-based search in the preferred site selector

Locale-based search in the preferred site selector has been fixed, allowing sites to be reliably found and selected by locale again.

(CMS-29048)

Fix Blueprint SearchConstants Value

Fixed Blueprint constant `com.coremedia.blueprint.cae.search.SearchConstants.FIELDS#CREATION_DATE` to lower-case value `creationdate` as used in the Solr schema.

(CMS-28592)

Studio Multi-Preview can Change or Disappear Depending on User Permissions

Multi-Preview settings retrieval failed to use the correct read permissions. This can lead to changing or even disappearing Multipreviews in Studio, depending on the permissions.

This behavior was fixed so that a consistent settings retrieval using the correct read permissions is guaranteed.

(CMS-28574)

Fixed a Bug in Cap Repositories Spring Configuration

By a regression, the bean definitions `workflowRepository` and `capListRepository` were present when the corresponding repositories were not connected. The fix makes these beans conditional on properties `repository.caplist.connect` and `repository.workflow.connect` respectively.

(CMS-28505)

Category parameter is taken for augmented products

If you have an augmented product and set some alternative placements and if the product is contained in multiple categories (e.g. beside the primary category it is also part of the e.g. “newarrivals-mens” category). If the PDP fragment is called with such an additional category parameter (not the primary category id) it is taken for augmentation (and not the primary category of the product).

(CMS-28100)

Changed the logic for the metric com.coremedia.Store.Connected

Changed the logic for the metric `com.coremedia.Store.Connected` to monitor the open connections to the database. This ensures that the state changes when the last connection is closed due to an error instead of waiting for a future connection request to encounter an error. Requires the number of minumum open connections to be set to at least 1 (`sql.pool.minConnections`). This is enforced by the code - values below 1 will be ignored and the minumum will be set to 1. As the server must open at least one connection to work, this won't change its operation. Relies on the connection validator to discover broken connections. If a more timely detection of connection issues is required, reduce the values of the properties `sql.pool.validatorInterval` and `sql.pool.validatorTimeout`.

(CMS-18211)

5. Upgrade Information

This chapter contains all information relevant for an upgrade to the specific release.

5.1 Upgrade to 2512.0.0

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

Minor API Change in WorkflowObject

The return type of `com.coremedia.cap.workflow.WorkflowObject#getWarnings` has been weakened from `List<WorkflowException>` to `List<CapException>`. Due to a CMS internal generics inconsistency the result list of `getWarnings` may contain `CapException` objects even in older CMS versions. This fix does not change the behaviour, but only clarifies the type. Custom code may have to be adjusted accordingly.

(CMS-29380)

Removed Built-in Search and Replace Workflows

The built-in workflow definitions `global-search-replace.xml` and `markup-search-replace.xml` have been removed. These definitions were used with the Site Manager, which has already been removed in CMCC 12.

(CMS-29293)

Configuration for Command Line Tool "publishall" Changed

Command Line Tool *publishall* was previously configurable with non-option arguments: `cm publishall [ -a [ -cq <query> ] ] [ -t <threads> ] <cmsiorurl> <cmsuser> <cmspwd> <masteriorurl> <masteruser> <masterpwd>`. With the introduction of gRPC as the communication protocol and the need for additional parameters, the configuration was adjusted and unified. Instead of non-option arguments, now all configuration has to be given as options.

The full option list for *publishall* has been extended as follows:

```
cm publishall [-a [-cq <query>]] [-t <threads>] [-v] [-?]
 [--user <user> --password <password> [--url <ior url>] [--grpc-endpoint
    <gRPC endpoint> --http-base-uri <base uri>]
    --mls-user <user> --mls-password <password> [--mls-url <ior url>]
    [--mls-grpc-endpoint <gRPC endpoint> --mls-http-base-uri <base uri>]]
```

Options *user*, *password*, *url* (only for CORBA communication), as well as *grpc-endpoint* and *http-base-uri* (both only for gRPC communication) define the connection to the *Content Management Server* as in other command line tools. They replace and extend the previously used command arguments `<cmsior url> <cmsuser> <cmspwd>`.

Note that instead of using options *url*, *grpc-endpoint,* and *http-base-uri* the corresponding values can also be given as environment variables *REPOSITORY_URL*, *SPRING_GRPC_CLIENT_CHANNELS_CAP_ADDRESS*, and *REPOSITORY_HTTPBASEURI*, respectively. Connection options given as query parameters of the IOR URL (tool parameter *url*) can also be specified with system properties (see *Deployment Manual*, section *Unified API Spring Boot Client Properties*). For gRPC, those properties are the only way for specification of connection options.

New options *mls-user*, *mls-password*, *mls-url* (only for CORBA communication), as well as *mls-grpc-endpoint* and *mls-http-base-uri* (both only for gRPC communication) now define the connection to the *Master Live Server*. They replace and extend the previously used command arguments `<masteriorurl> <masteruser> <masterpwd>`.

(CMS-28803)

Feeder caches results of Tika text extraction

Content Feeder and *CAE Feeder* now cache the results of Apache Tika text and metadata extraction for indexed blob values. This improves feeding performance, if content is re-indexed with unchanged blob value, or if the same blob value is used in multiple content items. The Blueprint configures a default cache capacity of estimated 10 MB heap memory using configuration property `cache.capacities.feeder.tika.heap` in `application.properties`. Because of that, Feeders require slightly more heap memory. Caching can be disabled by setting the cache capacity to 0.

(CMS-28599)

Support for personalization variants

In order to support personalization variants, a new meta-property `variant` was added to `Content` items. It is immutable once set during content creation. It is accessible through `isVariant()` and its inverse `isBaseline()`. For a baseline content item, multiple variants may be created. Variants are created with the same content type as their baseline. They are used for delivering A/B tests or specialized content for user segments.

For custom code, please check whether it can handle variant content items. Especially, check whether variants that are returned from methods like `getReferrersWithDescriptor(...)` break the analysis of complex content structures. In such cases, you can use the new methods `getBaselineReferrersWithDescriptor(...)` to avoid reaching variants accidentally, as these methods efficiently limit the returned collection to baseline content items. Similarly, instead of using `getContentsWithValue(...)` it might more robust to access observed properties using `getBaselineContentsWithValue(...)`. Similarly, the already deprecated class `ContentStringPropertyIndex` might not be appropriate, because it includes variants in its results.

For the sites service and its associated classes, personalization variants are generally ignored in returned collections, but accepted as input arguments. That is, you can use the sites service for determining the site of a variant or for deriving variants, but variants are not contained in methods like `ContentSiteAspect.getVariantsByLocale`. In the unlikely case that you need to include variants in this context, use the alternative methods like `ContentSiteAspect.getAllVariantsByLocale` or `ContentSiteAspect.getAllVariants`.

The variant feature only becomes accessible to editor if you install the *Native Personalization* plugin for *Studio*. Without the plugin, editor cannot create variants. Before doing that, all Content Servers must have been upgraded to a release that supports variants.

(CMS-28490)

CAE Feeder Renamed Cache Class for Internal Caching

The *CAE Feeder* uses in-memory cache keys for caching fragment values returned by Java method `com.coremedia.cap.persistentcache.PersistentCache#getCached`. The cache class of these internally used cache keys has changed from the generic cache class `java.lang.Object` to the new cache class `com.coremedia.cap.persistentcache`. The default cache capacity of 10000 entries is the same as the Blueprint's default for the previously used cache class. For details, see also section "Getting the Fragment Key Value from the Persistent Cache" in the *Search Manual*.

(CMS-28413)

Removed Deprecated Method in AssetDocumentBlobFeedablePopulator

Removed deprecated method `setMaxBytes` of Blueprint class `com.coremedia.blueprint.feeder.cae.assets.AssetDocumentBlobFeedablePopulator`. Use configuration property `feeder.blob.max-size` instead to limit feeding of blobs by their size.

(CMS-28375)

Removed Deprecated Classes ValidFromPropertyConverter / ValidToPropertyConverter

Removed deprecated Blueprint Java classes `ValidFromPropertyConverter` and `ValidToPropertyConverter` in package `com.coremedia.blueprint.caefeeder`.

(CMS-28374)

Removal: Deprecated Java API Package com.coremedia.cap.server.search

The following deprecated interfaces and classes of package `com.coremedia.cap.server.search` have been removed from the Java API:

- `com.coremedia.cap.server.search.SearchContext`
- `com.coremedia.cap.server.search.SearchException`
- `com.coremedia.cap.server.search.SearchService`
- `com.coremedia.cap.server.search.SearchServiceResult`

(CMS-27992)

Removal of XML Importer

The deprecated XML Importer has been removed. As a replacement, a custom Unified API client, the Ingest Service for CoreMedia Content Cloud Service or the serverimport tool should be used.

For your custom solution, you can continue using the “importer” service and user.

(CMS-27871)

Hardened Server- and XLIFF-Export for Invalid XML Control Characters

The output of `cm serverexport` as well as the resulting XLIFF from XLIFF-Export have been hardened to not output invalid XML control characters. If these characters made it into the XML/XLIFF-files parsers fail reading them afterward.

Invalid control characters range from `0x0000` (NUL) to `0x001F` (Information Separator One (US - Unit Separator)) except for `0x0009` (Horizontal Tab), `0x000A` (Line Feed) and `0x000D` (Carriage Return). On export, these control characters are now removed, so that they don’t block subsequent import.

One exception to this rule: For attributes `name` and `folder` of a given exported content item, URL-encoding is applied to them now, just as it is already done for internal representation (in link lists as well as in links in Rich Text and Structs).

Specification Level 1: With this change we introduce a “specification level” explicitly noted in files created by `serverexport` such as:

```
<CMChannel
  specification="1"
  folder="/Sites/Chef%20Corp%2E/United%20States/English/Navigation"
  name="Chef%20Corp%2E%20USA%20Home%20Page"
  xmlns:cmexport="http://www.coremedia.com/2012/cmexport">
  <!-- ... -->
</CMChannel>
```

Upgrade Information

In short, no upgrade steps are required. You may still want to check your contents for “illegal character use” in properties. Regarding the new Specification Level 1 for files generated by `cm serverexport`, `cm serverimport` can transparently handle any files created with tool versions before that.

Illegal Character Handling: You may want to check if your existing server export or XLIFF-export files contain any of these illegal characters. To align with the new behavior, you can just strip them from the corresponding files.

As part of this change this API has been extended:

- `hox.text.xml.XmlAttributeEscapingWriter`
- `hox.text.xml.XmlAttributeWriter`
- `hox.text.xml.XmlUnparser`

All of them now offer configuration options to:

- filter invalid XML values (disabled by default)
- encode control characters `0x0009`, `0x000A`, `0x000D` to their numeric entities (disabled by default).

The `XmlAttributeWriter` also got three extra `write(...)` methods that may now be used to prevent applying substituting characters for ignorable space text nodes. They are used, for example, by the `XmlUnparser`.

Regarding Specification Level 1: `cm serverimport` can import any files generated with a tool version prior to CMCC v13.2512.0.0. It will auto-detect the missing specification level and assume artificial “specification level 0” where no URL-decoding is applied to attributes `name` and `folder`. Thus, also here, any upgrade steps are optional.

If you require to import the XML data of `serverexport` with a `serverimport` version prior to CMCC v13.2512.0.0, you can set `serverexport` to output the corresponding format via `--specification 0` command line option.

Content Adaptations: Content export files that ship with CoreMedia Blueprint got upgraded to specification level 1. They must not be imported with any `cm serverimport` tool prior to CMCC v13.2512.0.0. Starting with CMCC v13.2512.0.0 `serverimport` tools will be aware of export files created with a newer version of `serverexport` and will deny importing those newer files.

Contents in VCS: If you have stored your project contents in VCS, you may want to re-export these files are apply these changes via Shell scripts:

- Add `specification="1"` to the root-element.
- URL-encode attributes `name` and the path-arcs of `folder` (including URL-encoding dots to `%2E`).

Tip: Despite handling URL-encoding of dots, you can use `jq --raw-input --raw-output @uri`:

```
echo "Chef Corp. USA Home Page" | jq --raw-input --raw-output @uri
```

(CMS-27456)

CAE Feeder Spring Configuration Changes and Deprecations

The Blueprint CAE Feeder Spring configuration was changed to use Spring Java configuration classes instead of legacy Spring XML files. This makes it possible to use Java code to configure which content bean data should be indexed. Instead of configuring content bean classes and properties as literal string values, the configuration can now use more reliable Java code to refer to methods directly. To this end, new `addElement` methods have been added in class `com.coremedia.cap.feeder.bean.BeanFeedableMapping` while API for old-style Spring XML configuration has been deprecated.

The new `BeanFeedableMapping#addElement` methods take a `java.util.function.Function` parameter to compute the values for Feedable elements. The function can also handle default values and converters, which were separate options of the formerly used class `BeanPropertyFeedableElementMapping`. For more information, see section "Configuring the BeanMappingFeedablePopulator" in the Search Manual, the API documentation or usages of class `BeanFeedableMapping` in Blueprint code. As part of this change, the class `BeanFeedableMapping` was changed to generic type `BeanFeedableMapping<T>` with type parameter `<T>` for the bean class, so that the source value of the value function is correctly typed.

Replaced Blueprint Spring XML and Properties Files:

- in module caefeeder-lib, `framework/spring/caefeeder-taxonomy.xml` replaced by `com.coremedia.blueprint.caefeeder.CaeFeederTaxonomyConfiguration`
- in module caefeeder-blueprint-component, `META-INF/coremedia/caefeeder.xml` replaced by `com.coremedia.blueprint.caefeeder.config.CaeFeederBlueprintAutoConfiguration`
- in module am-caefeeder-component, `META-INF/coremedia/am-caefeeder.xml` replaced by `com.coremedia.blueprint.feeder.cae.assets.AssetCaeFeederAutoConfiguration`
- in module lc-caefeeder-component, `META-INF/coremedia/livecontext-caefeeder.xml` replaced by `com.coremedia.blueprint.lc.caefeeder.LiveContextCaeFeederAutoConfiguration` and `META-INF/coremedia/livecontext-caefeeder-contentbeans.xml` for content bean definitions

- in module corporate-caefeeder-component, `META-INF/coremedia/corporate-caefeeder.xml` replaced by `com.coremedia.blueprint.caefeeder.corporate.CorporateCaeFeederAutoConfiguration`

The following properties files for Spring XML have been replaced by a Spring `ConfigurationProperties` class:

- in module corporate-caefeeder-component, `META-INF/coremedia/corporate-caefeeder.properties` replaced by `com.coremedia.blueprint.caefeeder.corporate.CorporatePageGridConfigurationProperties`

Custom changes in these files have to be migrated to Java classes as well, or to other custom configuration files. It is recommended to migrate all custom Spring XML files for the CAE Feeder to Spring Java configuration as well.

Replaced Framework Spring XML Files:

Some previously imported Spring XML framework files have been removed as part of this change. If you still import them in custom Spring XML files, you can replace them by importing Java classes or other Spring XML files as follows:

- For removed `classpath:/framework/spring/persistentcache/persistentcache-dependency.xml`, import Java configuration class `com.coremedia.amaro.persistentcache.PersistentCacheConfiguration` instead.
- For removed `classpath:/framework/spring/caefeeder/caefeeder-triggers.xml`, import Java configuration class `com.coremedia.amaro.cae.feeder.CaeFeederConfiguration` instead.
- For removed `classpath:/com/coremedia/blueprint/base/caefeeder/bpbase-caefeeder-pagegrid.xml`, import `classpath:/com/coremedia/blueprint/base/pagegrid/impl/bpbase-pagegrid-services.xml` and Java configuration class `com.coremedia.amaro.persistentcache.PersistentCacheConfiguration` instead.
- For removed `classpath:/com/coremedia/blueprint/base/caefeeder/bpbase-caefeeder-multisite.xml`, import Java configuration class `com.coremedia.blueprint.base.caefeeder.multisite.CaeFeederMultiSiteConfiguration` instead.

Configuration Property Changes:

Some properties for the configuration of page grid indexing (prefix `corporate.search.pageGrid`) have changed with the introduction of Spring Boot Configuration Properties in Blueprint class `com.coremedia.blueprint.caefeeder.corporate.CorporatePageGridConfigurationProper` `ties`. Property names have also changed to use kebab-case by default, but camel-case syntax also still works due to Spring Boot's relaxed property binding.

The following configuration properties have changed and do not accept space-separated values for multiple content types anymore. Instead, the configuration has to follow Spring Boot's way of configuring Map entries, with the content type as Map key being specified at the end of the property:

- Instead of `corporate.search.pageGrid.itemTextProperties`, use `corporate.search.page-grid.item-text-properties.<ContentType>`
- Instead of `corporate.search.pageGrid.inlineContentTextProperties`, use `corporate.search.page-grid.inline-content-text-properties.<ContentType>`

For details, see the Blueprint Developer Manual, section "Configuring page grid indexing" in chapter "Website Search".

Deprecated API:

- In class `com.coremedia.blueprint.base.caefeeder.pagegrid.PageGridFeedablePopulator`, the following methods have been deprecated: `setTextPropertiesAsString`, `setTextPropertiesByViewTypeAsStringValuedMap`, and `setRecursiveItemsPropertiesAsString`. This class was not part of the official public API in previous releases, but used in Blueprint Spring XML files. It is part of the documented public API now.
- In class `com.coremedia.cap.feeder.bean.BeanMappingFeedablePopulator`, the following methods have been deprecated: `getBeanPropertiesByClass` and `setBeanPropertiesByClass`.
- In class `com.coremedia.cap.feeder.bean.BeanFeedableMapping`, the no-argument constructor and the constructor taking a `Class` and `String[]` parameter have been deprecated.
- In class `com.coremedia.cap.feeder.bean.BeanFeedableMapping`, the following methods have been deprecated: `setBeanClass`, `getBeanProperties`, `setBeanProperties`, `getBeanPropertyMappings`, `setBeanPropertyMappings`, `getTextBody`, `setTextBody`, `getParameters`, and `setParameters`.
- Class `com.coremedia.cap.feeder.bean.BeanPropertyFeedableElementMapping` has been deprecated.
- Interface `com.coremedia.cap.feeder.bean.PropertyConverter` and its implementing public API classes `com.coremedia.cap.feeder.bean.CollectionPropertyConverter`, `com.coremedia.cap.feeder.bean.CollectionToStringPropertyConverter`, and `com.coremedia.cap.feeder.bean.ReplacePropertyConverter` have been deprecated.

See deprecation comments in the Java API documentation of the corresponding classes for details and how to replace custom usages.

In addition to these Java API deprecations, the Spring bean with name `caeFeederBeanPropertiesByClass` is also deprecated. The bean made it possible to configure bean classes and properties as string values, which made sense for usage in legacy Spring XML configuration files. It should not be used with Spring Java configuration classes anymore, which can reference classes and getter methods in a more robust way in code. To that end, use the Spring list bean `caeFeederBeanMappings` instead with one additional `BeanFeedableMapping` per bean class. For each such mapping, configure it by calling `BeanFeedableMapping#addElement(String, Function, boolean)` for each of its properties (comma-separated values in map value). To also index values in the `textbody` index field, as it was the case with the `caeFeederBeanPropertiesByClass` bean, use `true` for the parameter `textbody` of that method.

Deprecated Blueprint Classes:

The following Blueprint classes were also marked as deprecated. They can be deleted, if they are not used in custom code or Spring XML files anymore:

- `com.coremedia.blueprint.caefeeder.NavigationPathPropertyConverter`
- `com.coremedia.blueprint.caefeeder.PrefixedPathHierarchyConverter`
- `com.coremedia.blueprint.caefeeder.TaxonomyIdsPropertyConverter`
- `com.coremedia.blueprint.caefeeder.TaxonomyNamesPropertyConverter`
- `com.coremedia.blueprint.caefeeder.TaxonomyPropertyConverter`
- `com.coremedia.blueprint.caefeeder.NamedTaxonomyFactory`

(CMS-27398)

Removed cm groovysh

We removed the command line tool `cm groovysh` in favor of the more mature solution for ad-hoc Unified API scripting and more using `cm jshell` that we ship since CMCC v10.2004.1.

`cm jshell` is described in Content Server Manual in Section *Configuration and Operation / Server Utility Programs / Operation*. The tool provides additional help within JShell using the `coremedia` namespace:

```
$ cm jshell
jshell> coremedia.help()
```

(CMS-25885)

Deprecated Workflow JAR Upload Disabled by Default

The deprecated feature to upload JAR files with classes, and to use classes from previously uploaded JAR files in workflow process definitions has been disabled by default, and may be removed in a future release. Required classes should be added to the Workflow Server classpath instead.

If still needed, the possibility to upload and use JAR files with workflow process definitions can be enabled again by setting deprecated *Workflow Server* configuration property `workflow.server.use-uploaded-jar` to `true`. However, it is recommended to keep it disabled, also to avoid potential problems of storing executable code in the database.

(CMS-23931)

Removed Obsolete Property sql.store.login-user-name

The property `sql.store.login-user-name` has become obsolete and has been removed. Just use `sql.store.user` instead.

Be aware, that if you change the value of `sql.store.user` and are also using encrypted passwords in the database with `cm encryptpasswords` (see section “Encryptpasswords” in the Content Server Manual), the path to the key file `$INSTALL_DIR/etc/keys/<databasename>.<dbuser>.rijndael` will change. You can then either adjust the the name of the file to match the pattern again or you can configure the property `cap.server.encrypt-passwords-key-file`.

(CMS-23177)

Content Server Database Change

When the *Content Server* starts the first time after upgrading, it will automatically remove the no longer needed foreign key constraint `fk_WfV_group` on column groupValue_0 of table WfVariables for existing Oracle, MariaDB and MySQL databases.

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. This also the case for the otherwise unchanged database PostgreSQL. You should always make a backup of the Content Server database before upgrading. This is even more important for this change.

(CMS-18857)

5.1.2 CoreMedia Content Application Engine

Cache Control config is mapping fragment urls by default

In the Cache Control configuration URL patterns can be used to define a certain cache behaviour. These URL patterns can be extended to include matrix parameters. Matrix parameters are crucial to map fragment URLs in a commerce-led scenario. The default was changed to support matrix parameters by default.

(CMS-29071)

Image Transformation Operation: @Param.alias Changed to String-Array

@Param (FQN: com.coremedia.transform.dispatch.Param) got its parameter alias changed from type String to String[].

The new feature is used for a typo in the brightness operation parameter “amount” that was misspelled as “amout”. The definition is now rewritten to also still support “amout” if used in any stored transformation Strings:

```
@Operation(alias = "b")
void brightness(ImageTransformerState<Image> state,
                @Param(name = "amount", alias = {"a", "amout"}) double amount,
                @Param(name = "contrast", alias = "c", defaultValue = "1.0")
double contrast);
```

Upgrade Information

The type-change of `@Param.alias` does not affect any annotation usage, as former specifications like `@Param(name="op", alias="o")` will still continue to work.

If you processed `@Param` annotations though, you need to adapt your code.

(CMS-28410)

Removed Deprecated Template API

The following deprecated API and all their usages have been removed:

- com.coremedia.blueprint.common.layout.Container#getFlattenedItems
- com.coremedia.blueprint.common.contentbeans.CMMedia#getCaption

Along with this change, the content bean aspect API (package `com.coremedia.cae.aspect`) has been undeprecated.

The Blueprint content bean methods `getAspects` and `getAspectsByName` are no longer deprecated. All related deprecation annotations have been removed.

(CMS-27565)

CAE: Deprecated public api removed

The following deprecated API has been removed:

- `com.coremedia.cae.webflow.FlowRunner`
- `com.coremedia.blueprint.base.multisite.cae.SiteResolver#findSiteByPath`
- `com.coremedia.objectserver.web.ContentBlobHandlerBase`

See previous deprecations for intructions to replace usages.

(CMS-27562)

Replaced FreeMarker XML Spring Configuration with FreeMarker Spring Boot AutoConfiguration

CoreMedia's XML Spring configuration for the FreeMarker integration has been replaced with Spring Boot's `@AutoConfiguration` for FreeMarker.

The CAE now uses the `org.springframework.boot.autoconfigure.freemarker.FreeMarkerAutoConfiguration` to configure the FreeMarker integration. With this change, the integration now uses the standard configurations and extension points that are provided by Spring Boot and Free-Marker. Please check the following breaking changes and apply your project's configurations and customizations!

FreeMarker Settings

The FreeMarker settings have been changed to configuration properties. The `freemarker.` **configurations have to be changed to** **`spring.freemarker.settings.`** properties.

Additionally, the formerly non-standard setting `freemarker.template.update.delay.in.seconds` has been standardized to `spring.freemarker.settings.template_update_delay`.

Fo further information see Spring Boot Templating Properties and FreeMarker Settings.

Auto Imports

Custom auto imports are not configured with `@Customize("freemarkerConfigurer.autoImports")` or `<customize:append bean="freemarkerConfigurer" property="autoImports"/>` anymore. They are now added to the `spring.freemarker.settings.auto_import` property in the `application.properties` files for the CAE's, using the pattern `[templatePath] as [namespaceVarName]`. See also FreeMarker API: Configurable.

Shared Variables

Custom shared variables are not configured with `@Customize("freemarkerSharedVariables")` or `<customize:append bean="freemarkerSharedVariables"/>` anymore. They are now configured using `FreeMarkerVariablesCustomizer` beans:

```
@Bean
FreeMarkerVariablesCustomizer myFreemarkerSharedVariablesCustomizer() {
  return variables -> variables.put("myFreemarkerFacade", new
MyFreemarkerFacade());
}
```

Replaced XML configurations

Several Spring XML configuration files have been removed:

- `apps/cae/modules/cae/cae-base-lib/src/main/resources/framework/spring/blueprint-freemarker-views.xml`
- `apps/cae/modules/extensions/es/es-cae/src/main/resources/META-INF/coremedia/es-cae-views.xml`
- `apps/cae/modules/extensions/lc/lc-cae/src/main/resources/META-INF/coremedia/livecontext-freemarker-views.xml`
- `apps/cae/modules/extensions/lc/lc-cae/src/main/resources/META-INF/coremedia/livecontext-views.xml`
- `cae-viewservices.jar!com/coremedia/cae/view-freemarker-services.xml`
- `coremedia-common.jar!com/coremedia/image/image-service.xml`

Their former configurations are covered by the `FreeMarkerAutoConfiguration` or have been migrated to Java configurations.

(CMS-27279)

Profile "turbojpeg" Removed

Profile *turbojpeg* has been removed from the Blueprint workspace. It was experimental in the first place and unsupported for several years. Built-in image transformation in CAE and Headless Server continues to work as before. If in need for high-performance image transformation, consider using CoreMedia’s Image Transformation Service.

(CMS-22262)

5.1.3 CoreMedia Studio Client

Public API changes for Localization / Severity

The following types / consts / classes are now deprecated. You can find additional information about what to use instead in the code documentation:

- "Severity" of "@coremedia/studio-client.base-models"
- "ILocalization", "ILocalizationWithIcon", "ITypeLocalization", "IPropertyLocalization" and "IContentTypeLocalization" of "@coremedia/studio-client.cap-base-models"
- "IContentTypeLocalizationRegistry#getLocalizationByContentType" of "@coremedia/studio-client.cap-base-models"

For the interfaces "WorkflowIssueLocalization", "WorkflowIssuesLocalization" and "WorkflowLocalization" of "@coremedia/studio-client.workflow-plugin-models" we now have explicit configuration interfaces "WorkflowIssueLocalizationConfig", "WorkflowIssuesLocalizationConfig" and "WorkflowLocalizationConfig".

While the former is meant to provide a consistent, more strict access to registered Localization the latter is meant to provide configuration where some properties can be undefined or have shorthand syntax. The configuration interfaces are now also used when registering new workflows via the CustomWorkflow API.

This means 2 things:

- In case you have made use of the non-configuration interfaces inside your custom code, e.g. in function signatures of to store local variables, you might need to switch to the new configuration interfaces instead.
- You might be able to simplify code that utilizes localization interfaces as many properties can no longer be undefined and do have their shorthand syntax resolved to the explicit syntax.

(CMS-29309)

Third party updates

The version of many third party dependency was updated to their latest minor/patch versioning.

While we expect no problems with these updates, their is one notable change after updating the eslint dependencies regarding the format:

If the extends fragment contains more than one base class and the line is too long, the format of this multiline "extends" fragment changed. Now each base class is written in a new line.

The affected Studio-Client Source code was updated accordingly.

(CMS-29268)

Content Variants Always Open in the Same Studio Tab And Opening Studio Tabs is Asynchronous

We introduced the concept of certain "content item groups" that are always opened in the same Studio tab. For example. content items that are localized variants of each other (they have the same transitive master content) are now always opened in the same tab. It is no longer possible to open such variant-related content items in different tabs.

As a consequence, the whole procedure of opening content items in the Studio has become asynchronous. So for those cases, where you currently expect that a content is immediately opened in a new tab once you use one of the methods from below, you now have to await the corresponding Promise.

The methods that are affected, are all the methods from the WorkAreaTabManager and the ContentTabManager that open content items or entities. They now return a Promise.

(CMS-28797)

SearchContentLinkSuggester deprecated API removed.

The `SearchContentLinkSuggester` leaked private API. This API has now been removed. A replacement is not available as those functions were very specific and not mean to be part of the public API.

(CMS-28659)

PNPM: minimumReleaseAge is now set to one day by default

To reduce the risk of installing compromised packages, we now utilize the new minimumReleaseAge setting (introduced with 10.20.x) to delay the installation of newly published versions for 1 day. As in most cases, malicious releases are discovered and removed from the registry within an hour this is meant to protect us sufficiently while still being able to install fixes for other security issues in time.

For technical reasons we have excluded packages with the scope "@coremedia" and "@jangaroo" via the minimumReleaseAgeExcludes from falling into this delay.

As this is part of the blueprint you are free to adjust these settings to your needs.

(CMS-28594)

Node.js 24 and PNPM 10.24.0 Upgrade

- Node was upgraded to the latest LTS version 24.11.1
- PNPM has received a minor upgrade to 10.24.0

(CMS-28547)

Removed Animations

The .webm based Studio animations have been removed from the Studio and replaced by .svg files. All searches and empty states are affected by this, including the workflow app.

(CMS-28526)

Removed Tab-Expand

The configuration flag for the tab expand has been removed. All formular tabs are always shown now.

The TabExpandPlugin plugin has been removed too.

The design of the tabs has been changed to a button style.

(CMS-28434)

Studio: Deprecated `SearchService` replaced with `SearchService2`

The `SearchService` has been replaced with `SearchService2`. While the `SearchService` was based on Callbacks and Remote Beans the `SearchService2` is based on Promises.

(CMS-28424)

CKEditor5 has been updated to version 46.1.1

CKEditor5 has been updated to version 46.1.1 and may require an update of the `ckeditor5` dependency in custom plugins.

(CMS-28267)

Studio-Client packages for build-config, eslint-config and i18n-models now have their own release cycle

The versions of following packages are no longer matching the CoreMedia CMS Release version but have (similar to jangaroo packages) a dedicated release cycle:

- @coremedia/studio-client.build-config
- @coremedia/studio-client.eslint-config (old name: @coremedia/eslint-config-studio-client)
- @coremedia/studio-client.i18n-models

In case you are referring these packages without the "catalog:" protocol in your package.json files make sure to utilize the corresponding version mentioned in the catalog entry of apps/studio-client/pnpm-workspace.yaml

(CMS-27991)

Studio Client Workspace now employs the PNPM Catalog Protocol

The Studio Client Workspace now employs the usage of the PNPM catalog protocol for all dependencies in package.json of all packages. Using the catalog protocol makes it much easier and less error prone to maintain only one version per dependency in a workspace. For details refer to the original documentation at https://pnpm.io/catalogs

Changes:

- pnpm-workspace.yaml has been adjusted to manage the catalog via the corresponding entry
- catalogMode is set to prefer (see: https://pnpm.io/catalogs#catalogmode)
- All package.json files have been adjusted so external (non-workspace) dependencies are using the catalog protocol. This applies to:
  - dependencies
  - devDependencies
  - peerDependencies
- This means that dependencies inside all package.json files will either have “workspace:*” or “catalog:” as version.

Upgrade Notes:

- If you have added new external dependencies to packages we highly suggest to add them to the catalog as well
- Adjust added dependencies to existing package.json files accordingly
- Adjust your dependencies in newly added package.json files accordingly

(CMS-27980)

Deprecation: @coremedia/studio-client.client-core-impl and @coremedia/studio-client.cap-rest-client-impl

All modules exposed in @coremedia/studio-client.client-core-impl and @coremedia/studio-client.cap-rest-client-impl have been merged into @coremedia/studio-client.client-core and @coremedia/studio-client.cap-rest-client respectively keeping the path within the corresponding package intact. Please change usages (e.g. dependencies, imports) accordingly.

Examples

Old dependencies in package.json (assuming the migration to the catalog protocol has already been done):

```
{
  "dependencies": {
    ...
    "@coremedia/studio-client.cap-rest-client-impl": "catalog:,
    "@coremedia/studio-client.client-core": "catalog:",
    "@coremedia/studio-client.client-core-impl": "catalog:",
    ...
  }
}
```

New dependencies in package.json:

```
{
  "dependencies": {
    ...
    "@coremedia/studio-client.cap-rest-client": "catalog:,
    "@coremedia/studio-client.client-core": "catalog:",
    ...
  }
}
```

Old imports in example.ts

```
...
import { RemoteJobBase } from "@coremedia/studio-client.cap-rest-client-impl";
import { RemoteBean, ValueExpression } from
"@coremedia/studio-client.client-core";
import { RemoteBeanImpl } from "@coremedia/studio-client.client-core-impl";
...
```

New imports in example.ts

```
...
import { RemoteJobBase } from "@coremedia/studio-client.cap-rest-client";
import { RemoteBean, RemoteBeanImpl, ValueExpression } from
"@coremedia/studio-client.client-core";
...
```

Backward compatibility

When importing modules from their old location you will get the same import as when directly importing from the now location. This not only applies to code build inside the studio-client workspace but also applies to plugins and other precompiled code. This compatibility layer will be maintained for the whole CMCC13 lifecycle and removed afterwards.

This task is still marked as breaking as all blueprint packages have been adjusted accordingly.

(CMS-27952)

Include Updated Referenced Content in Publication Workflows

When starting a publication workflow in Studio there is now a checkbox to include updated referenced content - similar when starting a translation workflow. `PublicationWorkflowPlugin` and `DefaultStartPublicationWorkflowFormBase` also have the config parameter `hideDependentContentsStrategyChooser` which can be set to hide the checkbox.

(CMS-27789)

Public API changed

Public API Removal

@coremedia/studio-client.cap-rest-client

- CapList, CapListRepository and EncryptionService of are not public anymore.
- CapConnection: removed members getCapListRepository and getEncryptionService (were already marked as @private)

New Public API

@coremedia/studio-client.cap-base-models

- registerMyEditedContentFilter - allows registering filters for the contents appearing in “My Edited Content”
- observeMyEditedContents - allows observing the filtered contents of “My Edited Content”

Blueprint Adjustments

@coremedia-blueprint/studio-client.main.am-studio

- ShowAssetsInEditedContentsCheckboxBase now uses (new) registerEditedContentsFilter

(CMS-25612)

Upgraded eslint to v9

Eslint was upgraded to v9.

As this involves changing to the new flat config, we decided to rename our eslint packages and provide a createEslintConfig function, which helps to setup the config for the corresponding package type

- @coremedia/studio-client.eslint-config (old name: @coremedia/eslint-config-studio-client)
  - per default lints "src" and "test" folders with standard rules
- @coremedia/studio-client.ext.eslint-config (old name: @coremedia/eslint-config-studio-client)
  - per default lints "src" and "joounit" folders with extjs specific rules (usually more forgiving than standard rules)

The following steps are necessary to packages using eslint:

1) package.json

- Change the devDependency to the new package name (see above)
- Remove the file pattern from the eslint script.

2) eslint.config.(m)js

- Create the eslint.config.mjs file. You can also create a eslint.config.js if the package is already ESM package. However, ExtJS packages can never be ESM packages.
- Write the following content into the file, using the corresponding eslint-config package

```
import { createEslintConfig } from
"@coremedia/studio-client.ext.eslint-config";

export default createEslintConfig();
```

- For more fine granular control over folders that should be linted you can utilize the options parameter of createEslintConfig, e.g.:

```
import { createEslintConfig } from
"@coremedia/studio-client.ext.eslint-config";

export default createEslintConfig({ sourceFolders: ["src", "my-src"] });
```

3) .eslintrc

- In case you do not have made any adjustments you can delete the .eslintrc
- Otherwise you need to migrate the changes to v9 (https://eslint.org/docs/latest/use/migrate-to-9.0.0) and add them to the eslint.config.(m)js which can be done like this:

```
import { createEslintConfig } from "@coremedia/studio-client.eslint-config";
import { defineConfig, globalIgnores } from "eslint/config";

export default defineConfig([
  createEslintConfig(),
  {
    // your custom eslint config
  },
]);
```

4) Run the linter and fix reported problems

As this is a major upgrade of the standard eslint some rules sets have been added or changed.

Our goal was to find the balance between avoiding major breaking changes for the code base but still benefiting from these improvements, so we adjusted the rules if necessary. However, there might still might be some newly reported problems to fix.

(CMS-25598)

Old BlueprintDocumentTypes_properties is gone

The old resource bundle BlueprintDocumentTypes_properties was used mostly for the localisation for the content types and their properties. As there are already a new mechanism for the content type localisation the old resource bundle is now removed from blueprint.

(CMS-21880)

5.1.4 CoreMedia Studio Server

Removed deprecated APIs from modules cap-rest-service and bpbase-studio-rest

The following deprecated API has been removed and may require adapting usages:

Module `cap-rest-service`:

- Added public API package `com.coremedia.rest.cap.content.search`.
- Removed unused `com.coremedia.rest.cap.content.UrlModifier`.
- Removed `CapObjectFormat` and `QueryUriResolver` from the public API package `com.coremedia.rest.cap.content.search` .

Module `bpbase-studio-rest`:

- Removed public API package `com.coremedia.blueprint.base.rest.convert`.

(CMS-6803)

5.1.5 CoreMedia Workspace

Upgraded to Spring Boot 4 / Spring 7 baseline

The Spring dependencies were upgraded to the Spring Boot 4 / Spring 7 baseline. With these upgrades, some further dependencies had to be upgraded as well.

Upgrade Information

The following Spring dependencies were upgraded:

- Spring Boot → 4.0.0
  - Spring Boot 4.0 Release Notes
  - Spring Boot 4.0 Migration Guide
- Spring Framework → 7.0.1
  - Spring Framework 7.0 Release Notes
- Spring Security → 7.0.0
  - What’s New in Spring Security 7.0
- Spring Data BOM → 2025.1.1
  - Spring Data 2025.1 Release Notes
- Spring GraphQL → 2.0.1
  - Spring for GraphQL 2.0
- Spring Webflow → 4.0.0
- Micrometer → 1.16.1
- Reactor Project BOM → 2025.0.1

Numerous third-party dependencies have also been updated, some of the more noteworthy of which are the following:

- GraphQL Java → 25.0

- Hibernate ORM → 7.1.11.Final
- Hibernate Validator → 9.1.0.Final
- Jakarta Servlet API → 6.1.0
- Tomcat → 11.0.15
- JUnit Jupiter → 6.0.1

Most notable changes

Module Dependencies

Spring Boot 4.0 has a new modular design and now ships smaller focused modules rather than several large JARs. With this, many Spring Boot dependencies and Java packages have changed and respective references must be applied.

Take extra care of Spring Boot's auto configurations. Many of them have now vanished from the classpaths, because they were moved to module JARs that are not configured as project (compile) dependencies. Some of them weren't actually used, and they don't pollute the classpaths anymore. Others may be required for your application's functionality but now could have vanished. The new module JARs containing these auto configurations may have to be added as additional runtime or test dependencies now. To identify potentially missing auto configurations, you may start your Spring Boot applications or tests before and after upgrade with `-Ddebug=true` and compare the `CONDITIONS EVALUATION REPORT` in the logs.

Some unwanted auto configurations from the former Spring Boot JARs had to be excluded explicitly. Now their respective module dependencies have been removed so that those exclusions are not needed anymore:

- The `MongoAutoConfiguration` had to be excluded when it was in the combined `spring-boot-autoconfigure` dependency. Now it resides in `spring-boot-mongodb` that is not added to the applications anymore.
- The `FreeMarkerAutoConfiguration` had to be excluded when it was in the combined `spring-boot-autoconfigure` dependency. Now it resides in `spring-boot-freemarker` that is not added to the applications anymore.
- The `DataSourceAutoConfiguration` had to be excluded when it was in the combined `spring-boot-autoconfigure` dependency. Now it resides in `spring-boot-jdbc`. That dependency is still needed in the *Studio Server* and *User Changes* applications (along with the exclusion of the `DataSourceAutoConfiguration`). Bur it has been removed all other applications.

The former `ServletWebServerFactoryAutoConfiguration` and `EmbeddedWebServerFactoryCustomizerAutoConfiguration` contained configurations for various supported web servers. The configurations for each web server have now been moved to a focused module JAR. In case of Tomcat, the auto configuration class is now named `TomcatServletWebServerAutoConfiguration` and resides in the `spring-boot-tomcat` module JAR.

Upgrading Jackson

Spring Boot now uses Jackson 3 as its preferred JSON library. But for projects that cannot easily migrate, Jackson 2 is still supported (until excluding Spring Boot 4.2). Jackson 3 uses new group IDs and package names with `com.fasterxml.jackson` becoming `tools.jackson`. Although the coordinates of both Jackson versions are disjunct, it turned out that Spring's Jackson 2 and 3 integrations produce various conflicts when the dependencies of both versions are on the classpath of the CoreMedia applications. As migrating the Jackson integrations of the CoreMedia CMS is a rather complex and time-consuming task, Jackson 3 dependencies have been avoided for now.

The migration to Jackson 3 is planned for the AEPs that will be released in spring or autumn 2026. If possible, upcoming AMPs may support both Jackson versions on the classpaths, but this requires further investigation.

Upgrading Testing Features

Mocks or test clients may not be working like before. Test classes may require additional annotations now:

- `@AutoConfigureMockMvc`
- `@AutoConfigureWebTestClient`
- `@ExtendWith(MockitoExtension.class)`
  In some cases, when `@Mock` and `@MockitoBean` are both used in the same test class, the `@Mock` an

Public API changes

Spring Boot's new modular design required changes to the com.coremedia.cms.delivery.configuration.DeliveryPropertiesAutoConfiguration:

- The class annotations changed from
  {{@AutoConfiguration(after= {ServletWebServerFactoryAutoConfiguration.class,WebMvcAutoConfigura

)}} `@EnableConfigurationProperties(DeliveryConfigurationProperties.class)` to `@AutoConfiguration(after=WebMvcAutoConfiguration.class)` {{@EnableConfigurationProperties(

  {DeliveryConfigurationProperties.class,ServerProperties.class,WebMvcProperties.class}

)}}.

- The bean method changed from
  ```
  deliveryBaseURI(DeliveryConfigurationProperties, ObjectProvider<ServerP
  ```
  to
  ```
  deliveryBaseURI(DeliveryConfigurationProperties, ServerProperties, WebM
  ```

(CMS-28130)

Support and Require Maven 3.9.11

Supported and Required Maven version has been updated to 3.9.11.

Also, the Takari Maven Extensions have been removed (`.mvn/extensions.xml`) which means the option `--builder smart` is not available anymore by default.

See https://maven.apache.org/docs/3.9.11/release-notes.html

(CMS-27846)

5.1.6 CoreMedia Frontend Workspace

Updated Third Party Dependencies

Third party dependencies of the frontend workspace have been updated to their latest version fulfilling semantic version ranges.

(CMS-29269)

Frontend Workspace now employs the PNPM Catalog Protocol

The Frontend Workspace now employs the usage of the PNPM catalog protocol for all dependencies in package.json of all packages. Using the catalog protocol makes it much easier and less error prone to maintain only one version per dependency in a workspace. For details refer to the original documentation at https://pnpm.io/catalogs

Changes:

- pnpm-workspace.yaml has been adjusted to manage the catalog via the corresponding entry
- catalogMode is set to prefer (see: https://pnpm.io/catalogs#catalogmode)
- All package.json files have been adjusted so external (non-workspace) dependencies are using the catalog protocol. This applies to:
  - dependencies
  - devDependencies
  - peerDependencies
- This means that dependencies inside all package.json files will either have “workspace:*” or “catalog:” as version.

Upgrade Notes:

- If you have added new external dependencies to packages we highly suggest to add them to the catalog as well
- Adjust added dependencies to existing package.json files accordingly
- Adjust your dependencies in newly added package.json files accordingly

(CMS-29265)

Upgraded eslint to v9

Eslint was upgraded to v9.

As this involves changing to the new flat config, we decided to provide two functions, which help to setup the config for the corresponding package type

- `createFrontendEslintConfig`
  - per default lints "src" and "test" folders with standard rules
- `createFrontendToolsEslintConfig`
  - per default lints "src" and "test" folders with standard rules and adds additional globals for node

The following steps are necessary for packages using eslint:

1. `package.json`

- Remove the file pattern from the eslint script.

1. `eslint.config.(m)js`

- Create the `eslint.config.mjs` file. You can also create a eslint.config.js if the package is already ESM package.
- Write the following content into the file, using the corresponding eslint-config package

```
import { createFrontendEslintConfig } from "@coremedia/eslint-config-frontend";

export default createFrontendEslintConfig();
```

- For more fine granular control over folders that should be linted you can utilize the options parameter of createEslintConfig, e.g.:

```
import { createFrontendEslintConfig } from "@coremedia/eslint-config-frontend";

export default createFrontendEslintConfig({ sourceFolders: ["src", "my-src"]
 });
```

1. `.eslintrc`

- In case you do not have made any adjustments you can delete the .eslintrc
- Otherwise you need to migrate the changes to v9 (https://eslint.org/docs/latest/use/migrate-to-9.0.0) and add them to the eslint.config.(m)js which can be done like this:

```
import { createEslintConfig } from "@coremedia/studio-client.eslint-config";
import { defineConfig, globalIgnores } from "eslint/config";

export default defineConfig([
  createEslintConfig(),
  {
    // your custom eslint config
  },
]);
```

1. Run the linter and fix reported problems

As this is a major upgrade of the standard eslint some rules sets have been added or changed.

Our goal was to find the balance between avoiding major breaking changes for the code base but still benefiting from these improvements, so we adjusted the rules if necessary. However, there might still might be some newly reported problems to fix.

(CMS-25597)

5.1.7 CoreMedia Deployment

compose-wait command-line tool has been removed from the base image

To make distributed deployments of applications more robust, the java-application-base image contains support to wait for service dependencies. Up to now, this was done using the compose-wait tool. This tool has now been replaced with the much better fit wait4x. To use it the same way like compose-wait, there is an entrypoint script named `wait4x`, that can be included before the application script. See https://github.com/coremedia-contributions/docker.java-application-base for more details.

(CMS-27827)

confd has been removed from the java-application-base image

confd is no longer maintained and mutating files at startup should be avoided. Instead mount static config files if necessary.

(CMS-22475)

Switch studio-client and overview to CoreMedia vendored unprivileged nginx image.

The Docker images for studio-client and overview now use the CoreMedia vendored nginx unprivileged image. Port configuration and other adjustments have been applied to support the unprivileged setup. As a result, these containers no longer use port 80 and have been reconfigured to use port 8080. This change may require adjustments to your existing deployment or firewall settings."

(CMS-16467)

5.1.8 CoreMedia Commerce Hub

Renamed grpc.stubs Properties

The properties prefixed with `grpc.stubs` were renamed to a prefix of `grpc.client`. More specifically, the following properties are affected by this change:

- `grpc.client.default-timeout`
- `grpc.client.service-timeouts`

Rename the properties to conform to their new prefix.

(CMS-27833)

Commerce Hub gRPC Client Requires Proto Reflection V1

Commerce Hub gRPC client was updated to require Proto Reflection V1 on the commerce adapter side. Commerce Adapters using Base Adapter older than v3.0.11 are *not* compatible.

(CMS-27122)

5.1.9 CoreMedia Headless Server

Headless Server: Deprecated content type 'application/graphql+json' not supported anymore

With the necessary update to Spring 7, the already deprecated usage of `MediaType#APPLICATION_GRAPHQL` was removed in Spring 7, thus can’t be used anymore. The recommended replacement is `MediaType#APPLICATION_GRAPHQL_RESPONSE`.

Since the different media type values are used for the HTTP content type header, this means, that clients cannot use the old value `applica` `tion/graphql+json` anymore. Uses `application/graphql-response+json` instead!

If you're using `application/graphql-response+json` already, you don't need to change anything and might consider this as non-breaking.

(CMS-28677)

Headless now support image type transformation for original images (non-crop)

The media delivery of headless formerly offered image type conversation for predefined crops only. This change introduces the ability to transform also the original image.

Since this change updates aspects of the public API, it is considered breaking. However, these changes only affect the creation of two spring beans, namely `TransformationService` and `ResponsiveMediaHashCacheKeyFactory`. The responsible spring configuration was removed from public API. In most cases, this should not lead to any problems, unless you subclassed `TransformationServiceConfiguration` or `TransformationService` itself. Note that `TransformationService` is now a Java interface.

(CMS-28460)

Headless Server Content-Schema Deprecations in CM13

Starting with CM13, these deprecated fields were removed GraphQL Type Blob

```
size: Int
```

Please see also corresponding removals at `ContentBlobAdapter` and `ContentMarkupAdapter` in the public Java-API.

GraphQL Type CMMediaImpl

```
captionLegacy: String
```

```
captionAsTree: RichTextTree
captionReferencedContent: [Content_!]
```

Please use the property `caption: RichText` with its' generic, nested properties `text`, `textAsTree` and `textReferencedContent` for the deprecated fields instead.

GraphQL Type CMTeasableImpl

```

detailTextLegacy: String
detailTextAsTree: RichTextTree
detailTextReferencedContent: [Content_!]

teaserTextLegacy: String
teaserTextAsTree: RichTextTree
teaserTextReferencedContent: [Content_!]
```

Please use the properties `detailText: RichText}}and {{teaseText: RichText` with its' generic, nested properties `text`, `textAsTree` and `textReferencedContent` for the deprecated fields instead.

(CMS-28253)

Headless Server Java-API Deprecations in CM13

Starting with CM13, these Java-API deprecations were now applied Evaluation-Context (part of rich text transformation)

In class `com.coremedia.caas.richtext.stax.writer.intermediate.eval.EvaluationContext`, the incomplete convenience constructor was now removed. Please use the remaining main constructor instead.

The getter `public ContentRepository getContentRepository()` was removed. To get any content from the content repository please use the method `public Content getContent(String id)`.

CMGrammarRichTextAdapterFactory (part of rich text transformation)

The incomplete constructor was removed. Please use the main constructor instead.

ResponsiveMediaHashCacheKey

The incomplete constructor was removed. Please use the main constructor instead.

CoercingMap and CoercingRichtTextTree

Both classes now implement the non-deprecated versions of methods `serialize()`,`parseValue() }}and {{parseLiteral()` of the GraphQL-Java Interface `graphql.schema`.`Coercing` only.

ContentBlobAdapter and ContentMarkupAdapter

The deprecated method `getSize()` was removed from both classes. Please use `getSizeLong()` instead.

MD5 hashes calculated for Persisted Queries in Relay Query Map uses now SHA256 hashes

If persisted queries are stored on HeadlessServer in a relay query format, the hashes for all queries have to be recalculated as SHA256.

SearchServiceAdapter and SearchServiceAdapterFactory

Deprecated legacy classes `SearchServiceAdapter` and `SearchServiceAdapterFactory` were removed.

QueryListAdapter, QueryListAdapterFactory and search-schema.graphql

Deprecated constructors of QueryListAdapter with `offset` parameter were removed. Use method QueryListAdapter#getPagingResult instead for paged queries.

Deprecated fields `CMQueryList#pagedItems` and `CMQueryList#filteredPagedItems` have been removed from `search-schema.graphql`. Use `CMQueryList#itemsPaged` and `CMQueryList#filteredItemsPaged` instead.

AdapterSupplier<T extends Adapter> extends Supplier<T>

The non-standard method `to()` was now removed from interface `AdapterSupplier<T extends Adapter> extends Supplier<T>`. Please use the standard supplier method `get()` instead.

(CMS-27563)

Changed Cache Control Defaults

Cache Control default configuration changed so that the "cache forever" values were increased from a *max-age* of 180 to 365 days and enhanced with 1 hour of *stale-while-revalidate*, 1 days of *stale-if-error*. The blob delivery Cache-Control was also enhanced with the *immutable* directive. The Cache-Control config for Headless Server GET responses was enhanced with 30 seconds of *stale-while-revalidate* and 10 minutes of *stale-if-error*. The Cache-Control config for CAE fragment requests was set to a *max-age* of 2 minutes along with 30 seconds of *stale-while-revalidate* and 10 minutes of *stale-if-error*.

(CMS-27336)

Headless Server: Improved extensibility and flexibility of Blob delivery

With the release of CM13 the Blob delivery has been completely revised. The controller responsible for the Blob delivery is now configured by a number of Spring beans, which are all available as ExtensionPoints. For details please see the new chapter 'Customization of Media Endpoints' Headless Server Manual.

Other improvements:

The Blob delivery now supports image type transformation not only for defined crop names but for original image files also.

The filename property of CMDownload documents is now supported. This change applies to both URL generation and Blob delivery for CMDownload documents. For details please see chapter 'Adapter' in the Headless Server Manual.

(CMS-25427)

5.1.10 Miscellaneous

Deprecation of Adaptive Personalization

The integration of Adaptive Personalization is deprecated in favor of the new Native Personalization.

This includes extensions `p13n`, `alx-p13n`, `lc-p13n`, `sfmc-p13n`, and `sfmc-es-p13n`. These features are discontinued and will be replaced by the new Native Personalization solution.

(CMS-29078)

Removed unused code from UriComponentsHelper

Unused code has been removed from the `com.coremedia.objectserver.web.links.UriComponentsHelper`. This results in removed and renamed public methods.

Renamed:

- `appendOrOverride(UriComponents, UriComponents)` → `append(UriComponents, UriComponents)`

Removed:

- `appendOrOverride(UriComponents, UriComponents, boolean)`
- `override(UriComponents, UriComponents)`
- `prependPath(String, UriComponents)`
- `prependPath(UriComponents, UriComponents)`
- `prependPath2(String, UriComponents)`

See also CoreMedia Shared Middle API.

(CMS-28780)

Adaptive Personalization (p13n) Extensions Disabled By Default

The Adaptive Personalization extensions have been disabled by default in the Blueprint.

Following extensions have been disabled: `p13n`, `alx-p13n`, `lc-p13n`, `sfmc-p13n`, and `sfmc-es-p13n`. To enable them in your Blueprint, please run the following code in you blueeprint workspace:

```
mvn extensions:sync -Denable=p13n,alx-p13n,lc-p13n,sfmc-p13n,sfmc-es-p13n
```

The `p13n-dynamic-include` brick in the Frontend Workspace has been changed to work with our new “Native Personalization” solution and is still used in the example themes.

The Adaptive Personalization extension contains four doctypes: `CMSelectionRules`, `CMSegment`, `CMP13NSearch`, and `CMUserProfile`. Instances and references in the example content outside of Adaptive Personalization extensions have been removed. If you for some reason have instances of these doctypes without actually using Adaptive Personalization, you can either destroy these instances, enable the extensions again or just add the doctypes to your Blueprint.

(CMS-28763)

gRPC as an Alternative to CORBA Communication

For internal communication between CoreMedia CMCC components, gRPC has been added as an alternative to CORBA. Communication via CORBA is still available and the default. As long as you plan to stick with CORBA, no changes to your code or operations procedures are required. Should you want to switch to gRPC as your communications protocol between CMCC components, the following sections will provide information and pointer for additional details to do so.

In the course of adding gRPC as a communications protocol, however, breaking changes were introduced:

- The old workflow client API and its implementations have been removed. They were actually unused after removing the *Site Manager*. This API was deprecated with 2406.2.2 and 2506.0.2. This affects the classes in the following packages:
  - `com.coremedia.workflow`
  - `com.coremedia.workflow.common.util`
  - `com.coremedia.workflow.worklist`
- Public workflow server API has been moved. The Java packages of the affected classes didn't change, but they are now contained in Java artifact `com.coremedia.cms:cap-workflow-server` instead of `com.coremedia.cms:cap-workflow`. Consequently, they cannot be used in extensions other than for the workflow server anymore. For clients, please use the Unified API. Documentation for the moved classes used to be at CoreMedia Shared Common API [|https://documentation.coremedia.com/cmcc-12/artifacts/2506.0.0/javadoc/common/com/coremedia/workflow/package-summary.html] and can now be found at CoreMedia Workflow Server API .

Upgrade Information

With release 2512.0 of the CoreMedia system, the Workflow Server does not yet offer gRPC services to its clients. Activating gRPC on clients (e.g., Studio Server or command line tools) won't do any harm as long as these are of release 2512.0, too. *Future* clients, though, will fail to connect to that Workflow Server if gRPC is active on them.

Keep this in mind when operating a *mixed-release* CoreMedia system setup. In such scenarios (e.g., during upgrades), it is recommended to leave clients in their default protocol mode which is CORBA.

While servers always offer both protocols for clients to connect, clients can be configured to use *either* gRPC *or* CORBA. The protocol can be chosen freely on each start of a component. Currently, CORBA is the default protocol.

Note that Content Servers and the Workflow Server also contain client parts for internal communication, e.g., for internal repository access, publication, replication, etc. Choosing a protocol on a server therefore *does* make a difference although services *offered* by the servers to their clients always cover both protocols.

Information for Developers

See *Blueprint Developer Manual*, sections *3.2.2 Docker Compose Setup / Docker Compose Configuration* and *4.3 Build and Run the Applications / Internal Communication Protocol* on how to activate gRPC in a development setup. Refer to *Unified API Developer Manual*, section *4.1.1 Creating a Connection* for details on connection parameters for gRPC.

Information for Operations Engineers

See *Operations Basics*, section *4.4 Communication between the System Applications* for details on how to configure and secure gRPC communication in your environment.

Replicator and *Publisher* require specific configuration for gRPC.

For the *Publisher*, refer to *Deployment Manual*, section *3.2.3 Properties for the Publisher*.

For the *Replicator*, refer to *Deployment Manual*, section *3.2.5 Properties for Replicator Configuration*.

In a Nutshell

Adjust hostnames and ports in the following variables to your setup where necessary. Mind that there is no scheme to the gRPC addresses as those are not URIs.

To enable gRPC communication for a client or the *Workflow Server*, set the following environment variables on the respective process (command line tool, local start via Maven, Docker container, etc):

- `REPOSITORY_USEGRPC=true`
- `SPRING_GRPC_CLIENT_CHANNELS_CAP_ADDRESS=content-management-server:40165`

- `SPRING_GRPC_CLIENT_CHANNELS_WF_ADDRESS=workflow-server:40365` (not for *Workflow Server*; this setting is not in effect yet but may be added without harm)
- `REPOSITORY_HTTPBASEURI=http://content-management-server:40180`

To enable gRPC communication for the *Content Management Server*, set the following environment variables on the *Content Management Server* process:

- `REPOSITORY_USEGRPC=true`
- `PUBLISHER_TARGET_0_GRPC_ENDPOINT=master-live-server:40265` (repeat for each publication target)
- `PUBLISHER_TARGET_0_HTTP_BASE_URI=http://master-live-server:40280` (repeat for each publication target)

To enable gRPC communication for a *Replication Live Server*, set the following environment variables on the *Replication Live Server* process:

- `REPOSITORY_USEGRPC=true`
- `SPRING_GRPC_CLIENT_CHANNELS_CAP_ADDRESS=master-live-server:40165`
- `REPLICATOR_PUBLICATION_HTTP_BASE_URI=http://master-live-server:40280`

To enable gRPC communication for a *Master Live Server*, just set

- `REPOSITORY_USEGRPC=true`

To enable gRPC communication for command line tool *publishall*, see new command line options in *Content Server Manual*, section *3.13.3.5 Publishall* .

(CMS-28689)

CORS Configuration Updated

In case you have existing CORS configuration, change your config from `caas.cors.` **to `cors.`** according to the deployment manual.

(CMS-28541)

Removed redundant dependency management from blueprint parent POMs

Imports for Spring dependencies have been removed from the `dependencyManagement` sections from the `blueprint-parent` POMs.

They were redundant with imports in the `com.coremedia.cms:common-thirdparty-for-plugins-bom`. Furthermore they overrode some exclusions from that BOM and thus accidentally lead to inconsistencies in the application class paths.

This change is considered breaking, because it removes the version properties `spring-framework.version` and `spring-security.version` from the parent POMs. In the default blueprint, they were configured with the same version that are defined in the `com.coremedia.cms:common-thirdparty-for-plugins-bom` anyway. But customers who want the customize these versions now have to introduce a `dependencyManagement` for these dependencies theirselves.

(CMS-28515)

Removed util class InputInput

The class `com.coremedia.util.InputInput` has been removed. It's only usage in the content server's `hox.corem.server.media.jdbc.BlobStore`, dynamically determining the length of uploaded blobs, has been replaced with util classes from Apache Commons IO .

(CMS-28497)

Do not start MongoDB and Elastic Worker by default

Due to several changes (Removal of Google Analytics connectors, Disabling of Elastic Social by default, Changing Collaborative Component persistence to SQL) the Blueprint Docker deployment has been adjusted to not start MongoDB and Elastic Worker by default. To start them, please use the compose files `elastic.yml` and `development-elastic.yml`.

(CMS-28438)

Removed deprecated BaseFileSystemConfiguration from ContentHub API

The class `BaseFileSystemConfiguration` has been removed as it is not needed anymore and might cause confusion.

It only provided one bean: `ContentHubMimeTypeService`. This bean is now always available in the Studio Server application context. If your `ContentHubAdapter` is implemented as a plugin, please use `ContentHubBeansForPluginsConfiguration` to obtain it.

(CMS-28214)

Content hub adapters are no longer delivered as bundled plugins

In previous versions the content hub adapters for youtube and rss were bundled in the studio-server and studio-client. The plugins are no longer bundled and have to be added like other plugins in each project.

(CMS-28213)

CatalogPicture#getUrl() removed from template api

CatalogPicture#getUrl() must not be used from templates, therefore it has been removed from the template api.

(CMS-28177)

Removed Deprecated API: Predicates

The deprecated class `com.coremedia.common.util.Predicates` has been removed from the Java API.

(CMS-28146)

Removal of Analytics Connectors

The integration of Analytics Connectors based on Elastic Core has been removed, i.e. extension `es-alx` and parts of extension `alx`, including the analytics doctypes. Functionality for website tracking is still available and the corresponding documentation has moved to the CoreMedia Blueprint manual.

Removed features:

- Top N Lists including
  - Analytics doctypes: `CMALXBaseList (abstract)`, `CMALXPageList` and `CMALXEventList`
  - Contentbean implementations
  - Studio configuration forms
- Retrieval tasks and persistence based on the `ModelService` (module `bp base-es-alx-common`)
- Studio: generic configuration, performance widgets, deep links

These features are discontinued and will be replaced by an integrated solution.

Removal of the analytics doctypes (`CMALXPageList` and `CMALXEventList`) leads to problems in the Content-Server, if content of these types exist in the repository. Solution options:

- Recommended: Remove all content of type `CMALXPageList` and `CMALXEventList` as described in the documentation (Content-Server Manual/Deleting Content Types).
- If it is not possible to delete the content, reduced analytics doctypes can be added as custom doctypes:
  -
    ```
    <DocType Name="CMALXBaseList" Parent="CMDynamicList" Abstract="true">
    </DocType>
    <DocType Name="CMALXPageList" Parent="CMALXBaseList">
    </DocType>
    <DocType Name="CMALXEventList" Parent="CMALXBaseList">
    </DocType>
    ```
  - To prevent Studio users from creating/editing these doctypes, rights can be configured appropriately.

(CMS-28074)

Elastic Social (es) Extensions Disabled By Default

The Elastic Social extensions have been disabled by default in the Blueprint.

This means that after switching to the SQL Persistence for Projects and other editorial features, the Blueprint by default does not connect to a MongoDB.

Following extensions have been disabled: `es`, `lc-es`. To enable them in your Blueprint, please follow the instructions in section *"Enabling the Elastic Social Extension"* in the *"Blueprint Developer Manual"*.

The `es-controlroom` extension has been resolved to avoid confusion as it was no Elastic Social extension. It only contained a component scan for and a few dependencies which have now been added to the default Blueprint modules.

The `brick-elastic-social` brick in the Frontend Workspace has been left untouched and is still used in the example themes.

The Elastic Social extension contains two doctypes: `ESDynamicList` and `CMMail`. Instances and references in the example content outside of Elastic Social extensions have been removed. If you for some reason have instances of these doctypes without actually using Elastic Social, you can either destroy these instances, enable the extensions again or just add the doctypes to your Blueprint.

(CMS-28072)

Support for MySQL 8.0 and PostgreSQL 13 has been removed

Please update to supported versions MySQL 8.4 / PostgreSQL 16.

(CMS-27936)

The workflowconverter initscript has been removed from the workflow-server image

The recommended way to run the workflowconverter is either by running it in-place at application startup activated by the flag `workflow.server.en` `able-workflow-converter` or by running the converter tool as an initcontainer or manually, when the workflow-server is stopped.

(CMS-27806)

Confd has been removed from the management-tools image

Confd had been used as a convenience layer to reconfigure the tools based on environment variable based profiles. With confd it was possible configure arbitrary files using a templating mechanism.

Since confd is no longer maintained and because rendering configuration files at runtime is against the best practice of immutable container execution, mounting configuration files is the recommended approach in case custom configuration files are required.

(CMS-27753)

Middle Tier: Removal of Deprecated API

Upgrade Steps:

The following deprecated API has been removed and may require adapting usages:

Module `bpbase-cae-api`:

- `com.coremedia.blueprint.base.cae.web.taglib.ImageFunctions`
  The overloaded frontend `ImageFunctions#getImageLinkForAspectRatio` methods have bee
  The implementation of `com.coremedia.blueprint.cae.web.taglib.BlueprintFreemar

Module `bpbase-lc-common`:

- `com.coremedia.blueprint.base.livecontext.ecommerce.common.CommercePropertyHelper`:
  - The method `replaceTokens(String, Map<String,Object>, boolean):String` has b
    Use `AbstractPropertyResolver#resolvePlaceholders(String):String` instead us

Module `bpbase-pagegrid-api`:

- `com.coremedia.blueprint.base.pagegrid.PageGridContentKeywords`
  - Removed constants `LEGACY_LINK_PROPERTY_NAME` (original value: `"LinkPropertyName"`) and `LEGACY_LINK_LIST_PROPERTY_NAME` (original value `"LinkListPropertyName"`) that were only required to deal with legacy content items. While corresponding legacy content items continue to work, if they define these properties, new page grids should use the values of `PageGridContentKeywords#SECTION_PROPERTY_NAME` for the former and `PageGridContentKeywords#ITEMS_PROPERTY_NAME` for the latter.

Module `bpbase-sfmc`:

- `com.coremedia.blueprint.base.sfmc.libservices.dataextensions.SFMCDataEx`
  The method `pushToDataExtension(SFMCContext, String, List<DataExtensionEnt`

Module `cae-contentbeanservices`:

- `com.coremedia.objectserver.blob.BeanPropertyBlob`
  - Aligned with deprecation removal `Blob.getSize():int` in common tier, removed method `getSize():int`. Use `getSizeLong():long` instead.

Module `cap-feeder-api`:

- `com.coremedia.cap.feeder.FeedableWeightHelper`
  - Removed method `weighElement(FeedableElement):int`. Use `weighElementLong(FeedableElement):long` instead.
  - Removed method `weighFeedable(Feedable):int`. Use method `weighFeedableLong(Feedable):long` instead.

Module `cap-rest-service`:

- `com.coremedia.rest.cap.validators.AvailableLocalesConfiguration`
  - Class removed. Related removals:
    - `com.coremedia.rest.cap.CapRestServiceBaseConfiguration#availableLocalesConfiguration`
    - `com.coremedia.rest.cap.validators.AvailableLocalesValidator#setConfi`
      Use constructor `AvailableLocalesValidator(String, String)` instead.
    - No-argument constructor `com.coremedia.rest.cap.validators.AvailableLocalesValidator`.

Module `cap-transform`:

- `com.coremedia.cap.transform.ExtractDimensionCacheKey`
  - Removed 5-arg constructor (with additional MIME-Type and size) as the additional parameters were ignored and the 3-arg constructor (signature: `ExtractDimensionCacheKey(ContentRepository, ImageDimensionsExtractor, String)`) should be used instead.
- `com.coremedia.cap.transform.ImageVariantSetting`
  - Removed methods `setGamma(Double):void` and `getGamma():Double` without replacement. They showed misaligned and thus hardly predictable behavior for different types of images.
- `com.coremedia.cap.transform.TransformImageService`:
  - The method `getTransformationString` has been removed without replacement (accidentally made it to public API).

Module `cap-unified-api-util`:

- `com.coremedia.cap.util.ObjectCacheKey`
  - Class removed. Use `com.coremedia.cache.util.ObjectCacheKey` instead (module: `coremedia-cache`).
- `com.coremedia.cap.util.PairCacheKey`
  - Class removed. Use `com.coremedia.cache.util.PairCacheKey` instead (module: `coremedia-cache`).
- `com.coremedia.cap.util.StructUtil`
  Methods removed. Use alternatives in `com.coremedia.cap.util.CapStructUtil` instead:

  - `findString(Struct, String):Optional<String>`
  - `getBoolean(Struct, String):boolean`
  - `getInteger(Struct, String):Integer`
  - `getLink(Struct, String):Content`
  - `getLinks(Struct, String):List<Content>`
  - `getLong(Struct, String):Long`
  - `getString(Struct, String):String`
  - `getStrings(Struct, String):List<String>`
  - `getSubstruct(Struct, String):Struct`
  - `getSubstructs(Struct, String):List<Struct>`
  - `hasPropertyDescriptor(Struct, String):boolean`

Module `coremedia-transform`:

- `com.coremedia.transform.image.ImageOperations`
  - Removed method `crop(ImageTransformerState<Image>, int, int, int, int, String):void` from the interface. Method to use and implement instead: `crop(ImageTransformerState<Image>, int, int, int, int, String):void`.

  - Removed method `gamma(ImageTransformerState<Image>, double):void` without repl
    Note on image transformations: With this change the image transformation operation `gamma` (alia

  - Replaced method `unsharpMask(ImageTransformerState<Image>, float, float, int):void` by its no-parameter variant `unsharpMask(ImageTransformerState<Image>):void`, which is aligned with the deprecation notice, that any other than the default parameters (amount=0.15, radius=2, threshold=10) should not be used.
- `com.coremedia.transform.image.java2d.Java2DImageOperations`
  - Adapted according to changes in `ImageOperations` (removed one of the implemented `crop` methods, method `gamma`).
  - Updated implementation of `unsharpMask` to align with the signature change applying the defaults for amount, radius, and threshold as described above.

Module `middle.cap-validation`:

- `com.coremedia.rest.cap.validation.ContentTypeValidatorBase`
  - Class removed. Use `com.coremedia.rest.cap.validation.AbstractContentTypeValidator` instead (same module).
- `com.coremedia.rest.cap.validation.PredicateAwareValidatorBase`
  - Class removed. Implement `com.coremedia.rest.cap.validation.CapTypeValidator` directly instead (same module).
- `com.coremedia.rest.cap.validators.CrossSiteLinkValidator`
  - Removed method `setSeverity(Severity):void`. Use `setDefaultSeverity(Severity):void` instead.

Module `middle.coremedia-validation`:

- `com.coremedia.rest.validation.ObjectPropertyValidatorBase`
  - Class removed. Use `com.coremedia.rest.validation.AbstractPropertyValidator` instead (same module).
- `com.coremedia.rest.validation.PropertyValidatorBase`
  - Class removed. Use `com.coremedia.rest.validation.AbstractPropertyValidator` instead (same module)

(CMS-27564)

Common Tier: Removal of Deprecated API

The following deprecated classes and methods have been removed:

- `com.coremedia.blueprint.base.navigation.context.ContextStrategy#selectContext`
- `com.coremedia.cache.CacheCapacityConfigurer`
- `com.coremedia.cap.common.Blob#getSize`
- `com.coremedia.cms.common.plugins.beans_for_plugins.CommonBeansForPluginsConfiguration`
- `com.coremedia.common.logging.MarkerFactory#getMarker`
- `com.coremedia.dispatch.InterfaceOnlyPredicate`
- `com.coremedia.livecontext.ecommerce.asset.AssetService#getDefaultPicture`
- `com.coremedia.livecontext.ecommerce.common.CommerceIdProvider#formatProductTechId`
- `com.coremedia.livecontext.ecommerce.common.CommerceIdProvider#formatProductVariantTechId`
- `com.coremedia.livecontext.ecommerce.common.CommerceIdProvider#formatCategoryTechId`
- `com.coremedia.livecontext.ecommerce.user.UserContext#getCookieHeader`
- `com.coremedia.livecontext.ecommerce.user.UserContext.Builder#withCookieHeader`
- `com.coremedia.xml.UnsafeJdkJaxpFactory`
- `com.coremedia.xml.XmlUtil5#escape`

Constructors

- `com.coremedia.cap.content.events.ContentCreatedEvent`
- `com.coremedia.cap.content.events.ContentDestroyedEvent`
- `com.coremedia.cap.content.events.ContentRevertedEvent`
- `com.coremedia.cap.content.events.VersionDestroyedEvent`

(CMS-27557)

Third-Party Update: Apache Solr 9.10.0

Apache Solr has been updated to version 9.10.0 and the coremedia/solr-base Docker image with tag 2.6-cm-9.10.0 has been released. As part of this change, the following transitive dependencies of Apache SolrJ have been updated:

- Apache Zookeeper 3.9.4

- Eclipse Jetty 10.0.26

The update fixes security vulnerabilities CVE-2025-5115 and CVE-2025-12383 in Solr.

Solr 9.10.0 includes an update of Lucene, and indices written by the new version may become unreadable for previous Solr versions. You should make a backup of Solr indices before the update to be able to revert the update in case of unexpected problems. To avoid errors with Solr replication, followers should be updated at the same time or before leaders, or Solr replication should be stopped during upgrade, for example by disconnecting Solr followers from the Solr leader.

The configuration property `solr.use-http1` for CMCC applications is no longer deprecated. It is still recommended to keep it at its default value `false` to use HTTP/2, if possible.

Methods to set the Solr request handler at the SolrQuery object have been deprecated in SolrJ (`SolrQuery#setRequestHandler` and `CommonParams.QT`). For CMCC 13 (2512.0.0), the Blueprint code has been adapted to set the request handler as path when executing the query (in package `com.coremedia.blueprint.cae.search.solr`).

See https://solr.apache.org/docs/9_10_0/changes/Changes.html and https://solr.apache.org/guide/solr/latest/upgrade-notes/major-changes-in-solr-9.html#solr-9-10 for a list of changes in Solr.

(CMS-27540)

Discontinue base image flavors

The java-application-base image no longer provides jdk based image flavors. The jre for corretto and temurin is an adjusted custom jre based on the default jre module set with the addition of jshell.

(CMS-27499)

Switched to spring-grpc

Switched from grpc-spring to spring-grpc. This change causes that properties below `grpc.server` and `grpc.client` are no longer relevant. Instead the properties below ``spring.grpc.server}}and {{spring.grpc.client}}must be used respectively. Along with this change, the commerce hub client setting {{com.coremedia.blueprint.base.livecontext.client.settings.CommerceSettings#getEndpoint}}was removed. Use `{{com.core`` `media.blueprint.base.livecontext.client.settings.CommerceSettings#getEndpointName}}instead and configure an ops-friendly symbolic name, which is then configured as a named gRPC channel below {{spring.grpc.client.channels`. More specifically, this means that your `commerce}}struct's {{endpoint}}value is no longer picked up to establish a commerce connection. Introduce a String property named {{endpointName}}in your {{commerce}}struct and give it the value of a gRPC channel, such as {{hybris}}or {{sfcc`.

(CMS-27495)

Java 21 Support

Starting with CMCC v13.2512.0.0 the required JDK to build the workspace and run the applications is Java 21 LTS Release.

Upgrade Notes

- **Maven Plugins have been updated.**
  This includes especially a required update of the `maven-dependency-plugin`. It should be noted t

- **Java Application Base Image updated.**
  A new Java Application Base Image is used for the CMS applications: `coremedia/java-applicati`

- **Removed Deprecated Java API Usages.**
  Deprecated API usages have been removed, which includes, but is not limited to:

  - `Locale` constructor: Changed to using `Locale.of()`.
  - `URL` constructor: Replaced by alternatives like `new URI(...).toURL()`.

- **Widened `@cm.template.api` annotations.**
  To address issues in the generated Javadoc for the template API, fixed missing `@cm.template.api` a

(CMS-27319)

Removal of deprecated Commerce Hub Data properties

The following deprecated properties have been removed. Use their replacements in parentheses '(…)' instead.

- `commerce.hub.data.deadline` (`grpc.client.default-timeout`)
- `commerce.hub.data.services-without-timeout` (`grpc.client.service-timeouts`)

The property `commerce.hub.data.endpoints` has been removed without direct property replacement. Instead, use the `endpointName` of the `commerce` settings struct (respectively `com.coremedia.blueprint.base.livecontext.client.settings.CommerceSettings#getEndpointName`) to configure the Commerce Hub connection.

(CMS-27298)

Removed Deprecated API for Custom Proactive Engine Application

Custom Proactive Engine / Persistent Cache applications are not supported anymore. Proactive Engine and Persistent Cache are internal technical components of the *CAE Feeder* that cannot be used in custom applications outside of the *CAE Feeder* anymore. The Maven artifact `com.coremedia.cms:proactiveengine-component` and deprecated public Java API have been removed.

The following deprecated classes and interfaces have been removed:

- `com.coremedia.cap.persistentcache.InvalidationEvent`
- `com.coremedia.cap.persistentcache.InvalidationListener`
- `com.coremedia.cap.persistentcache.ProcessingCallback`
- `com.coremedia.cap.persistentcache.ProcessingCallbackBase`
- `com.coremedia.cap.persistentcache.example.HelloPersistentCacheKey`
- `com.coremedia.cap.persistentcache.example.HelloPersistentCacheKeyFactory`
- `com.coremedia.cap.persistentcache.proactive.BaseReceiver`
- `com.coremedia.cap.persistentcache.proactive.IllegalStateTransitionException`

- `com.coremedia.cap.persistentcache.proactive.KeyManager`
- `com.coremedia.cap.persistentcache.proactive.KeyPrefixFilteringReceiver`
- `com.coremedia.cap.persistentcache.proactive.Receiver`
- `com.coremedia.cap.persistentcache.proactive.TriggerBase`
- `com.coremedia.cap.persistentcache.proactive.content.ContentAwarePersistentCacheKeyFactory`
- `com.coremedia.cap.persistentcache.proactive.content.ContentBeanEvaluator`
- `com.coremedia.cap.persistentcache.proactive.content.ContentEvaluator`
- `com.coremedia.cap.persistentcache.proactive.content.ContentPersistentCacheKeyFactory`
- `com.coremedia.cap.persistentcache.proactive.content.ContentTrigger`
- `com.coremedia.cap.persistentcache.proactive.example.LoggingReceiver`
- `com.coremedia.cap.persistentcache.proactive.example.PathContentEvaluator`

The following interface of package `com.coremedia.cap.persistentcache.proactive.content` has been removed with replacement:

- `ContentTrigger.ContentCollection` - use `ContentSelection` instead

The following methods of package `com.coremedia.cap.persistentcache.proactive.content` have been removed with replacement:

- `ContentSelector.start(ContentTrigger.ContentCollection)` - use method `start(ContentSelection)` instead.
- `PathAndTypeContentSelector.start(ContentTrigger.ContentCollection)` - use method `start(ContentSelection)` instead.

The methods `startEvaluators` and `stopEvaluators` of class `com.coremedia.cap.persistentcache.proactive.ProactiveEngineManagement` have been removed. JMX operations of MBean `ProactiveEngine` with the same name have also been removed. They did not have any effect for the *CAE Feeder*.

(CMS-27251)

Studio-Client Docker build improvements

The Studio-Client Docker build has been simplified to provide a better convenience when building the image. All three Dockerfiles (Dockerfile, Dockerfile.tooling and Dockerfile.tasks) have been merged into a single Dockerfile which is then preconfigured using a Docker Bakefile.

The support of extensions at the root workspace path via archiving to zipfile has been replaced using an additional Docker context to the extensions directory. Please read the README.md d on how to enable the extension context.

(CMS-26030)

Support for Microsoft SQL Server (MSSQL) has been removed

As announced, the support for the database Microsoft SQL Server (MSSQL) has been removed from the product.

(CMS-25905)

Change Storage Layer of Collaborative Components to SQL

The Collaborative Components (My Edited Content, Projects, Workflow Archive, Notifications) now store their data in an SQL database instead of MongoDB. MongoDB and the In-Memory replacement are not supported any longer for these features, but can still be enabled for instance to prepare a data migration. For instructions on data migration and further details please see the section *Collaborative Components* in the *Operation Basics* Manual.

(CMS-25582)

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:

https://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

Altes Klöpperhaus, 5. OG

Rödingsmarkt 9

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.

Documentation

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.

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.

Support request

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.

Support checklist

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

Log 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.