

WHAT'S NEW

CoreMedia Content Cloud v11
CMCC Distribution 11.2201.1



CoreMedia Content Cloud Distribution 11.2201.1

The new product features and enhancements highlighted in this document are available as part of CoreMedia Content Cloud v11, distribution 11.2201.1. For more details about the included products and components, please refer to our distribution download site at <https://go.coremedia.com/cmcc-11>.

1.	HTTP Cache-Control for Headless Server	3
2.	Salesforce Commerce Cloud Cartridge Recertified.....	3
3.	Workflow Content Merged List.....	4
4.	Layout Variant Specific Queries.....	5
5.	Folder Properties in Studio	5
6.	Document Form Locale Switcher	7
7.	Updated Developer Preview of New Rich Text Editor.....	7
8.	Support for CoreMedia Content Cloud v11 (11.2110.x) in CMCC-Service.....	8
9.	Authentication with Azure AD in CMCC-Service	9
10.	Personal Access Token Management for CMCC-Service API Keys in Cloud Manager.....	10
11.	CMCC-Service API: Environment Information of Prod and PreProd Environments.....	11
12.	CMCC-Service API: Deployment of PreProd Environments	11
13.	CMCC-Service API: Commerce Cache Invalidation	11



1. HTTP Cache-Control for Headless Server

Websites and e-Commerce integrations always need to strike a balance between efficiency and accuracy. To address this challenge, CoreMedia has introduced a new API and default implementation to ensure that developers and administrators now have more control over the caching behavior of GraphQL content delivered by CoreMedia Content Cloud. HTTP Caching improves the website performance by instructing CDNs and clients to reuse previously fetched resources.

With the new CoreMedia Cache Control API and default implementation, projects have full control over caching behavior of GraphQL content delivered by the Headless Server.

This new feature consists of 1) configuration of URL patterns and default values, 2) an automatic mode that respects validFrom and validTo information in the content that is used during processing of a request. This ensures that the response is not cached longer than the next editorially planned date and time.

This new feature only applies to GET requests, e.g. (Automatic) Persisted Queries or REST-mapped queries. GraphQL POST requests do not support Cache-Control.

The benefits of the new Cache Control API include:

- More fine-grained control over caching of GraphQL responses.
- Ensures that planned editorial content changes will be reflected near time.

2. Salesforce Commerce Cloud Cartridge Recertified

Salesforce Commerce Cloud provides access to certified cartridges from partners like CoreMedia. Developers and administrators want to be sure to use a certified cartridge that complies with the most up-to-date Salesforce Commerce Cloud API standards.

With version 2201.1, CoreMedia provides an updated and re-certified CoreMedia LINK cartridge for Salesforce Commerce Cloud to version 21.1.0.

This cartridge is adapted to OCAPI Version 21.9 and compatible with Salesforce Reference Architecture (SFRA) 6.0.0. Smaller enhancements.

A certified version of the cartridge can be retrieved from SFCC by registered customers. It can be used for both CMCC10 and CMCC11.

3. Workflow Content Merged List

In previous versions of CoreMedia Content Cloud, content items that are part of a publication workflow have always been displayed in two distinct lists: 1) one containing documents the editor had actively added to the workflow, and 2) a second one containing necessary documents that are linked items in the first list but have not yet been published.

There was feedback that this separation into two lists was hard for Studio editors to understand.

With version 2201.1, these two lists have been merged into one – with an indicator showing content that has been automatically added to the workflow to guarantee link integrity.

This new list layout is now part of the Workflow App. It is utilized in the workflow detail view as well as in the workflow split view in the Content App.

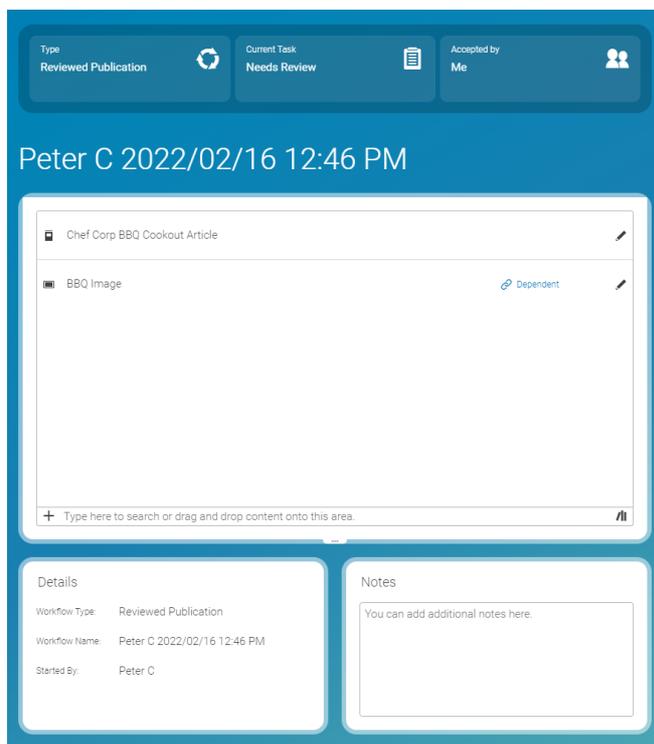


Figure 1: Updated workflow content list

This new list display approach provides a better overview of the relevant information regarding the current state of content in workflows, and will, hopefully, results in less confusion for editors.



4. Layout Variant Specific Queries

Front-end developer must be able to query data depending on view-types to avoid over-fetching.

For example, to position the teaser text properly with the hero view-type, it is necessary to also load the overlay configuration. However, this data is not required and should not be retrieved for a portrait banner.

To address this issue and increase system performance, it is now possible to formulate more precise queries to fetch the exact data required for the current view-type.

This capability is now provided as part of the Headless Server Graph-QL API.

This feature reduces the amount of bandwidth required and ensures a faster response times for requests against the Headless Server.

5. Folder Properties in Studio

In previous versions of CoreMedia Content Cloud, non-administrative users, who have supervisory rights, lacked the ability to manage rules/rights of folders in CoreMedia Studio. With the new "Folder Properties" dialogue in Studio, they can now edit the rules/rights for folders without using the SiteManager. Additionally, this new dialogue provides additional information, such as the folder ID, for every Studio user.

The new Folder Properties dialogue can now be accessed via the context menu of a folder. This dialogue provides two tabs:

- The "Filing" tab contains information, like the ID, creator, path, last modified and type
- The "Permissions" tab allows supervisory users and admins to edit the rights of this folder

The following screenshots show how this new dialogue will appear to users.

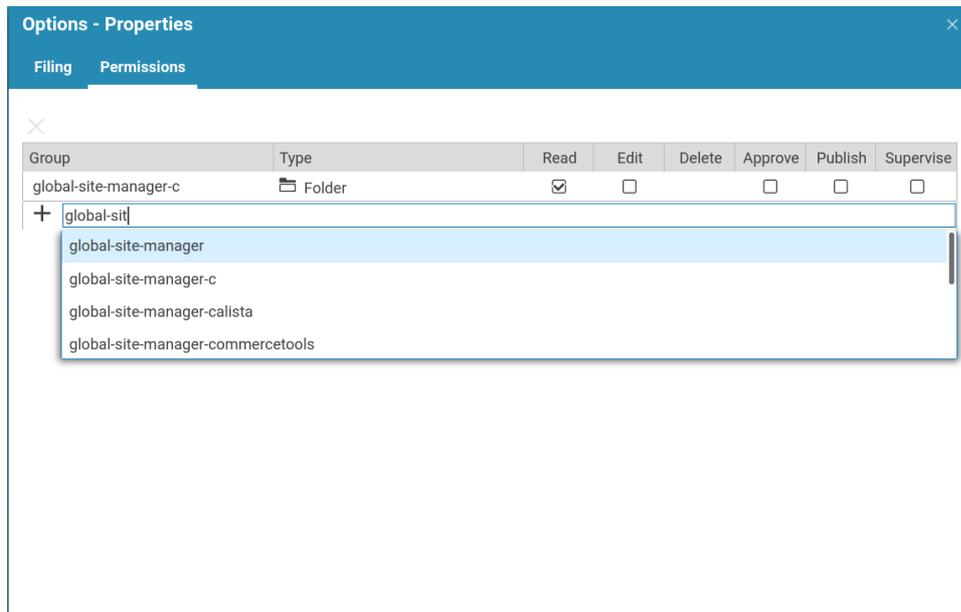


Figure 2: Folder properties dialogue permissions tab

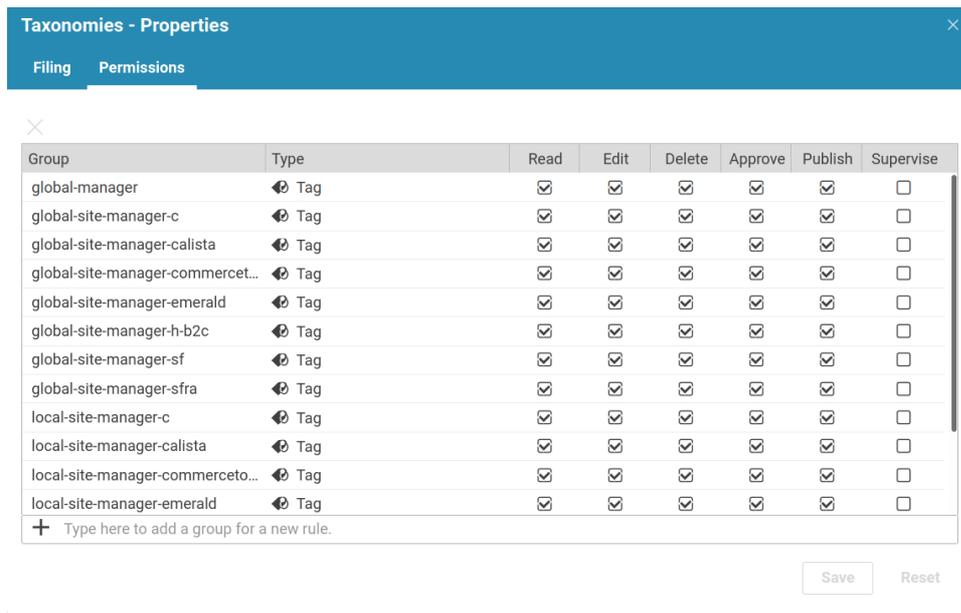


Figure 3: Editing permissions via the Folder properties dialogue



6. Document Form Locale Switcher

For Studio editors that work with different locales of a site, checking the translation/synchronization status of a content item, or even maintaining it, has always been quite cumbersome. With the new Document Form Locale Switcher, it is now possible to easily switch between the locales of a content item and maintain their translation/synchronization status as needed.

The Document Form Locale Switcher is accessed via the Form Toolbar of each document. The user gets a list of all locales available for this document and can switch to each of them. He also sees the translation/synchronization status to the master of the document and can update it by triggering a translation/synchronization workflow.

The following screenshots demonstrate how this looks in practice

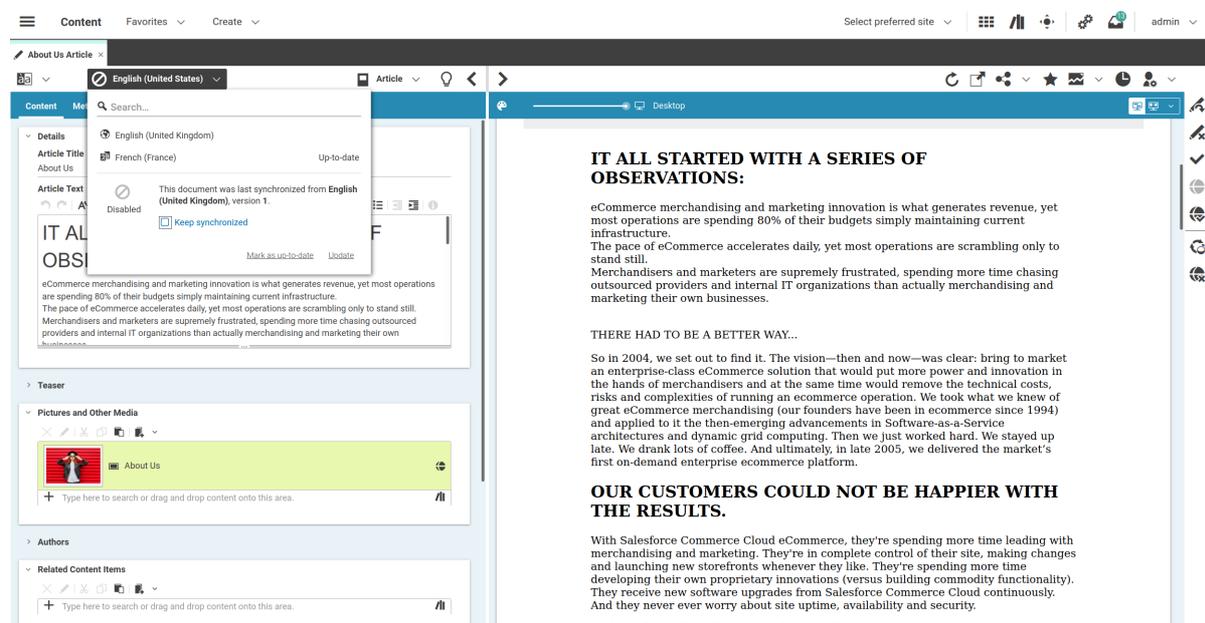


Figure 4: Document Form Locale Switcher example

7. Updated Developer Preview of New Rich Text Editor

CoreMedia is developing a new Rich Text editor that will be based on the modern and fully customizable CKEditor 5 framework, which provides numerous plugins to boost of editorial efficiency. The integration with CoreMedia Rich Text will be a standard CKEditor plugin itself. This should simplify the usage and development of other plugins.

With the GA release of CMCC v11, the first version of the Rich Text Editor preview was published to give developers an early look of what will be required to migrate custom CKEditor plugin and extensions to

the new editor. With each new CMCC release we will include a new version of the CKEditor 5-based Rich Text editor to keep developers up to date with the latest version so that they can provide early feedback.

With the new release we have added the following features to the developer preview (not to be used in production):

- General Rich Text Support to ensure that existing valid Rich Text is not discarded even if there is no corresponding plugin that enables editors to edit the specific markup in Studio.
- Drag and Drop support in Studio to enable use cases such as dropping images directly into Rich Text
- Updated to newer CKEditor 5 release with numerous fixes and improvements

The developer preview as an opt-in feature for the Studio client is part of the Blueprint workspace. The Open-Source GitHub repository serves as the starting point for the development of custom plugins with a concise documentation and sandbox environment.

To enable the preview, the Studio Client must be updated. The Studio developer manual contains instructions on how to accomplish this.

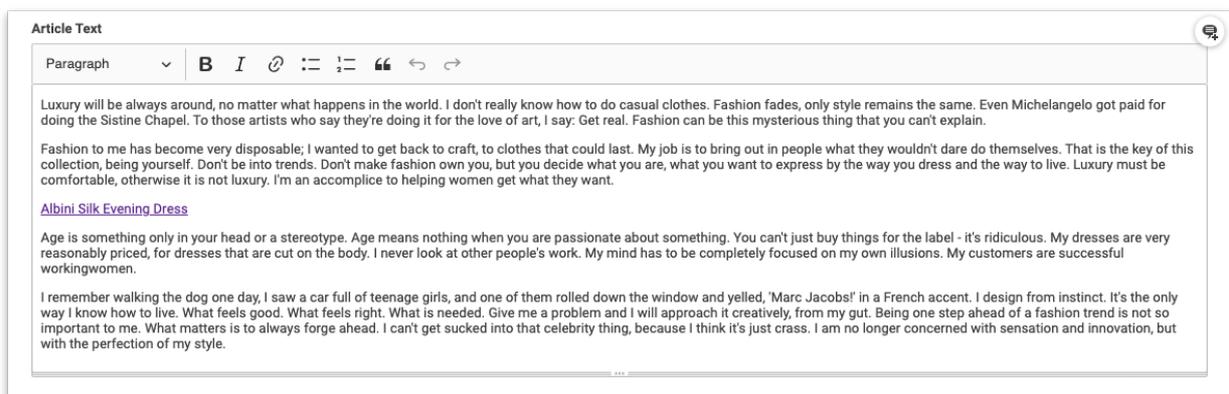


Figure 5: Developer preview or new Rich Text editor

Follow our development by watching our Open-Source GitHub repository (see <https://github.com/CoreMedia/ckeditor-plugins/>).

8. Support for CoreMedia Content Cloud v11 (11.2110.x) in CMCC-Service

CoreMedia Content Cloud v11 (11.2110.x) will be supported with the upcoming CMCC-Service release.

All existing CMCC–Service customers will be able to upgrade to CoreMedia Content Cloud v11 (11.2110.x). New customers will start their service on CoreMedia Content Cloud v11 (11.2110.x).

Support for CoreMedia Content Cloud v11 will be based on the same conditions and limitations as before:

- No Elastic Social
- No SiteManager
- No remote plugins (only bundled)

9. Authentication with Azure AD in CMCC–Service

Up to now, CMCC–Service customers have been limited to the use of CoreMedia’s native identity management capabilities for authenticating their users. With the upcoming CMCC–Service release, a customer’s Azure AD (Active Directory) can be configured to authenticate the users for the CMCC–Service.

With the feature enabled, a Cloud Manager user can choose between logging in with their Azure AD or internal authentication credentials.

To enable the CMCC–Service connection to their Active Directory, customers need to provide a configuration process described in the CMCC–Service manual.

Customers benefit from this new feature because they don’t have to manage their users twice (in CMCC–Service and Azure AD). They also profit from the enhanced security measures, such as MFA (Multi Factor Authentication), that are available through Azure AD.

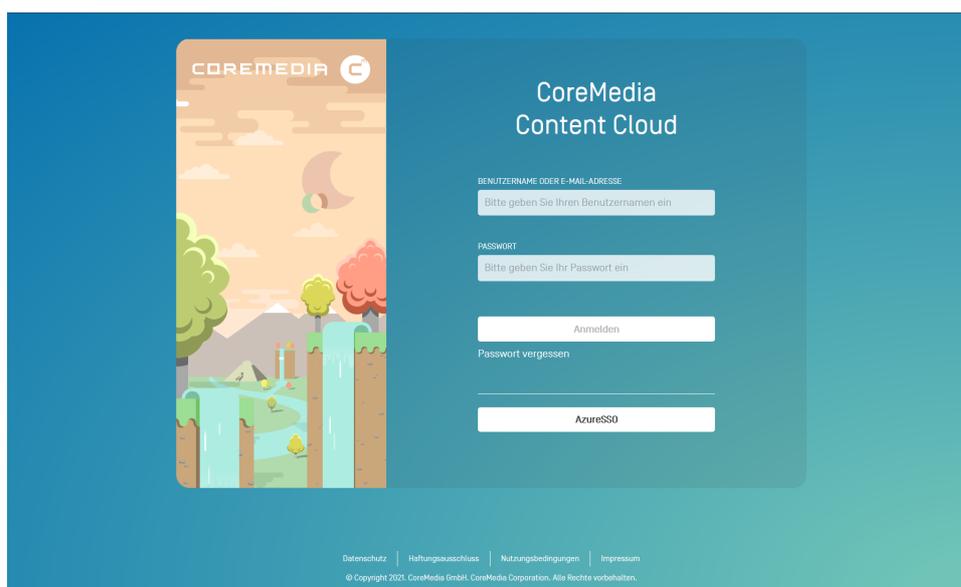


Figure 6: Cloud Manager Login Screen with Azure AD Login activated.



10. Personal Access Token Management for CMCC-Service API Keys in Cloud Manager

With the upcoming CMCC-Service release, customers will be able to use features of the CMCC-Service through a REST API, which enables them to build their own clients for their specific use cases.

To provide CMCC-Service administrators with a secure means of interacting with the API, we have introduced the management of personal access tokens within the Cloud Manager.

CMCC-Service administrators can create personal access tokens of specific rights which can be used to authenticate and authorize calls against the CMCC-Service API. CMCC-Service customers can then automate tasks in a secure way, which was previously only available through the Cloud Manager UI, or not at all.

The screenshot displays the 'PERSONAL ACCESS TOKENS' management interface in the Cloud Manager. On the left is a navigation sidebar with the user 'obauer' and the CoreMedia logo. The main content area is titled 'PERSONAL ACCESS TOKENS' and contains the following elements:

- ADD A PERSONAL ACCESS TOKEN** (link)
- Text: "Enter all details and we will return a unique personal access token. Administrators have access to all personal access tokens and can view their scopes and name."
- Name** (input field)
- Select Scopes** (header)
- Text: "Please select at least one permission for the personal access token."
- Production** (section header)
- Scopes (checkboxes):
 - content_read**: Read content via the Ingest Service.
 - content_write**: Read and write content via the Ingest Service.
 - deployment_read**: Read information about the running applications.
- UAT** (section header)
- Scopes (checkboxes):
 - content_read**: Read content via the Ingest Service.
 - content_write**: Read and write content via the Ingest Service.
 - deployment_read**: Read information about the running applications.
 - deployment_write**: Right to execute deployments.

Figure 7: Cloud Manager Personal Access Token Section



11. CMCC–Service API: Environment Information of Prod and PreProd Environments

CMCC–Service customers need information about their Prod/PreProd environments. With the new updates to the CMCC–Service API, it's now possible to query those environments for configuration details.

With the new CMCC–Service REST API endpoint, it is possible for technical CMCC–Service users to query configuration details about the running applications, such as the software version (Docker image tag) or certain environment configuration settings.

The use of the environment information REST API requires a valid personal access token with the 'deployment_read' right.

The benefit of this new feature is that customers can now query detailed human and machine-readable information about their environments:

- without the need to ask our support
- to support automation use cases.

12. CMCC–Service API: Deployment of PreProd Environments

Previously, CMCC–Service administrators and developers can trigger deployments in PreProd environments through a UI in Cloud Manager. With the new release, customers can now use a REST API to trigger such deployments.

This new feature consists of a self–service REST API for deployments of new software releases in PreProduction environments.

The usage of the deployment REST API requires a valid personal access token with the 'deployment_write' right.

The benefit of this feature is that customers can now integrate PreProduction deployments in their CI/CD infrastructure.

13. CMCC–Service API: Commerce Cache Invalidation

CoreMedia Commerce Adapters cache the shop data of their commerce system for use in their CMS clients (Studio, CAE).



Service administrators for CMCC–Service customers with a commerce integration need to invalidate the cache of the running Commerce Adapters in situations where case shop data needs to be updated immediately.

By default, this data is stored for a certain amount of time before it is fetched again. With the new version of CoreMedia Commerce Cloud, users can now invalidate this cache to proactively force the usage of actual shop data through a REST API endpoint (e.g., when a new campaign is published).

The usage of the commerce cache invalidation REST API requires a valid personal access token with the 'commerce_cache_invalidation_write' right.

The benefit of this feature is that customers can now force the update of commerce data dependent content as a self–service.