WHAT'S NEW

CoreMedia Content Cloud v10
CMCC Distribution 10.2010.1

COREMEDIA

CoreMedia Content Cloud Distribution 10.2010.1

The new product features and enhancements highlighted in this document are available as part of CoreMedia Content Cloud v10, distribution 10.2010.1. For more details about the included products and components, please refer to our distribution download site at https://l.coremedia.com/cmcc-10.

1. New Commerce Connectors

1.1. commercetools Connector

Over the last few years, CoreMedia Content Cloud has integrated with every major eCommerce system on the market. And now we're introducing a new integration with commercetools.

Headless commerce and content have gained significant traction over the last few years – and with good reason. A headless frontend layer enables developers to focus just on styling, independently from the backend software. It allows easy switching of technologies without the need to revamp a complete stack. And it separates the frontend from backend technologies, making it easier to employ specialists to manage each part efficiently.

One of the major vendors for headless commerce is commercetools, a headless, API-first, multi-tenant SaaS commerce platform that is cloud-native and uses flexible microservices. With distribution 10.2010.1, commercetools is now integrated with CoreMedia Content Cloud on a product-level.

CoreMedia customers who rely on commercetools to manage their online shops can now enable marketers, merchandisers and developers to collaboratively create, preview, and digital brand experiences that combine digital content and commerce elements.

CoreMedia's commercetools Connector includes a ready-to-use CoreMedia Commerce Hub Adapter that can connect to any commercetools project. This integration connects commercetools with the CoreMedia Studio as well as all CoreMedia content delivery components, including the Content Application Engine (CAE) for server-side rendering and the CoreMedia Headless Server for client-side rendering.

This connector allows commercetools editorial users to:

> Augment existing store pages and create new hybrid pages that combine rich media and branded content with product information and transactional capabilities
> Create personalized stories around any product or category
> Preview the entire online shopping experience on any device, for any customer segment, across the entire customer journey

Developers that build the storefront can use the CoreMedia Headless Server to fetch content to dynamically augment commerce sites with banners, carousels, shoppable videos, image maps, editable text on images, slide shows, 360 spinners, and much more.

For customization purposes, CoreMedia provides a Github repository with a custom workspace.
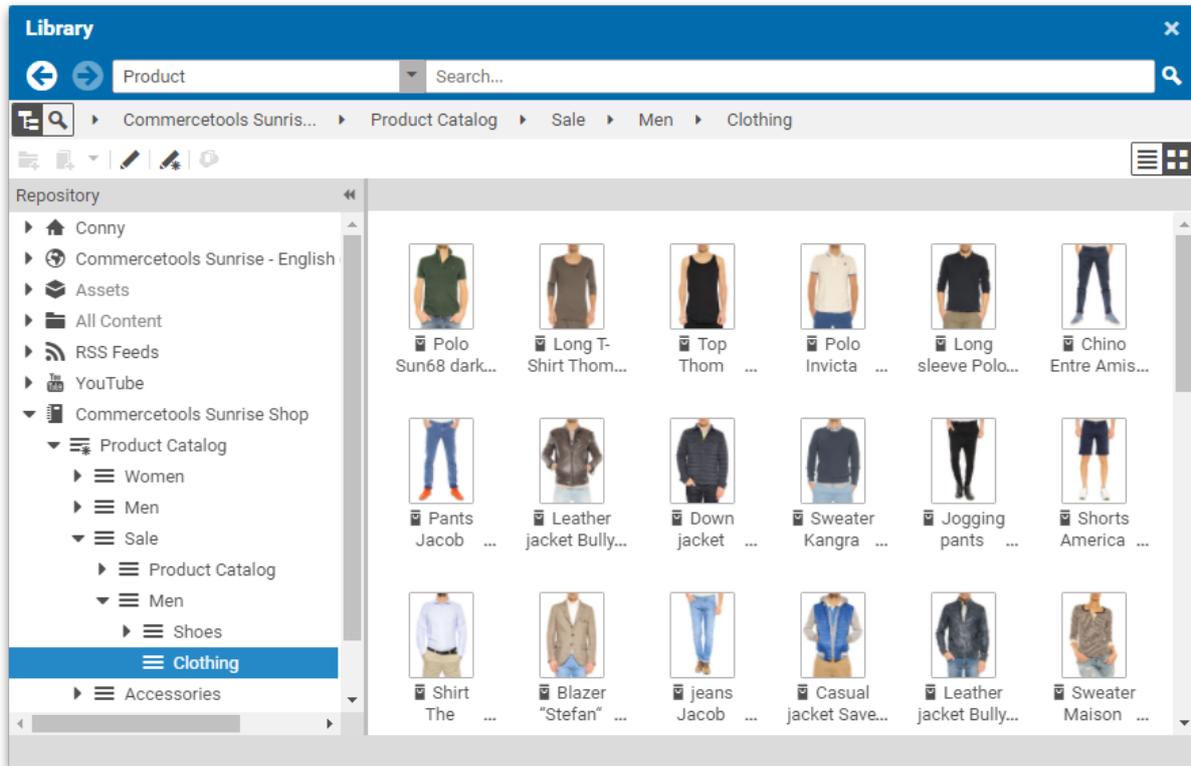
Access the commercetools product catalog directly from CoreMedia Studio
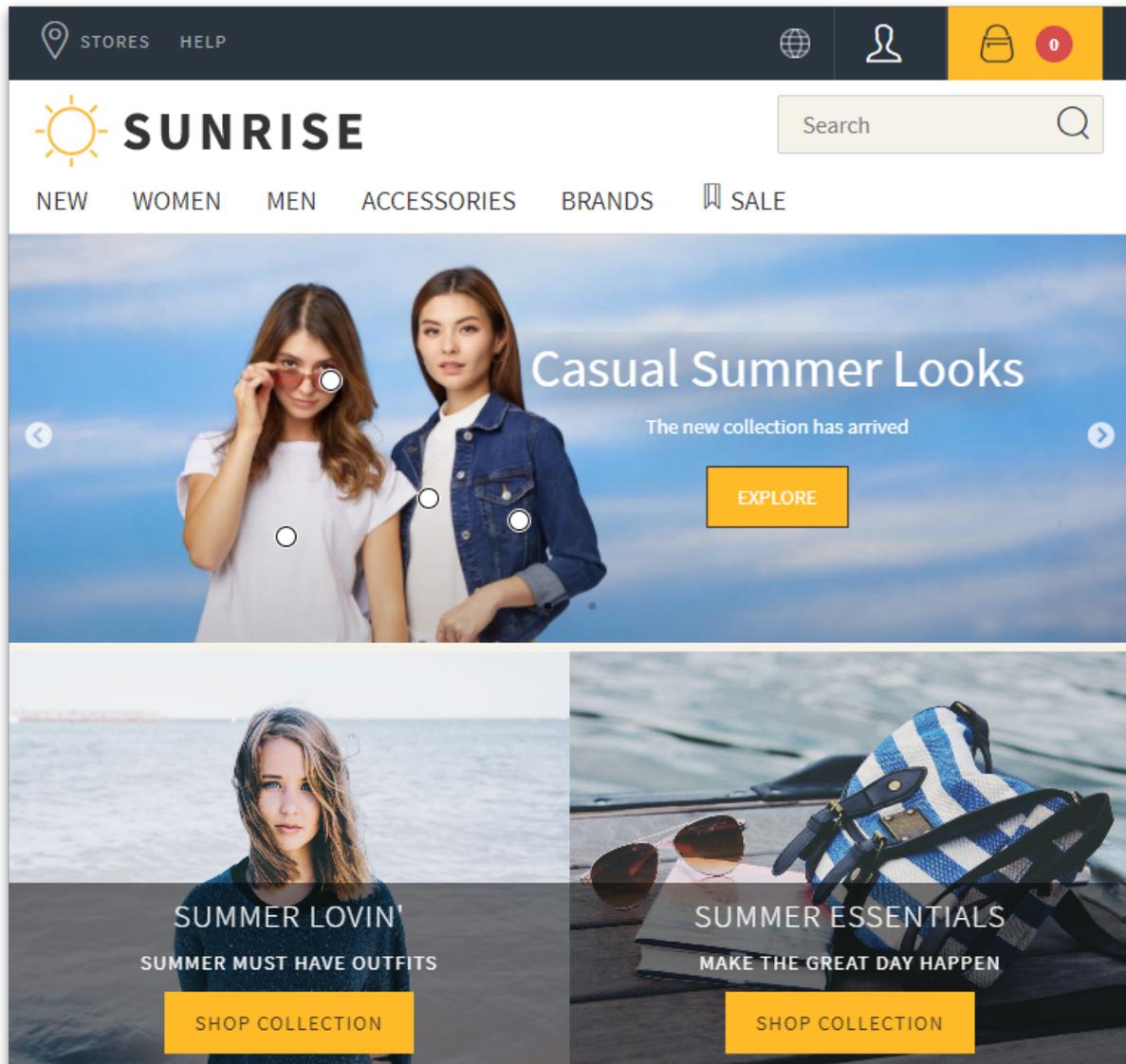
Augment commercetools storefront

Because CoreMedia supports a wide range of integration patterns for content, commercetools customers can use this connector:

> To deliver content fragments or data into an existing online store (commerce-led or headless)
> To render the entire experience (content-led approach)
> For anything in between (hybrid approach).

Because the commercetools client is based on Vue and uses the Apollo framework to load data from a GraphQL endpoint, frontend integration with CoreMedia is relatively straightforward. User can

simply add a second endpoint pointing to the CoreMedia Headless Server and start loading content and data into their Vue client immediately.

The hybrid headless approach of this integration provides all the flexibility and support that developers demand, plus advanced tools for marketers and merchandisers to quickly preview, edit and optimize content in context and in real-time.

1.2. SAP Commerce Cloud 2005 Connector

SAP Commerce Cloud is designed to deliver "next-generation business modeling" that simplifies complex commerce and order processes. But limited options exist for creating powerful content experiences. That's where CoreMedia Content Cloud comes in. Our product-level integration allows SAP Commerce Cloud customer to build next-generation storefronts and content-rich, shoppable brand sites by blending real-time product information with multimedia marketing content at every stage in the customer journey.

SAP Commerce Cloud 2005 is the latest major release for on-premise SAP Commerce Cloud customers. CoreMedia Content Cloud distribution 2010 provides out-of-the-box support for this latest version.

CoreMedia's Commerce Hub connector for SAP Commerce Cloud 2005 enables editorial users to browse the catalog within CoreMedia Studio, augment existing store pages with rich media and interactive content, target personalized content, and manage assets for products and categories from the latest SAP Commerce Cloud 2005. Developers that build the storefront can use our custom workspace to enable content fragments to be shown in the SAP Commerce storefront and integrate the tag-library into their JSP templates.

This feature consists of:

> a ready-to-use Commerce Hub Adapter that can connect to SAP Commerce Cloud 2005
> A workspace with SAP Commerce Cloud extensions and examples.
> For customization purposes, CoreMedia provides a GitHub repository with a custom workspace.
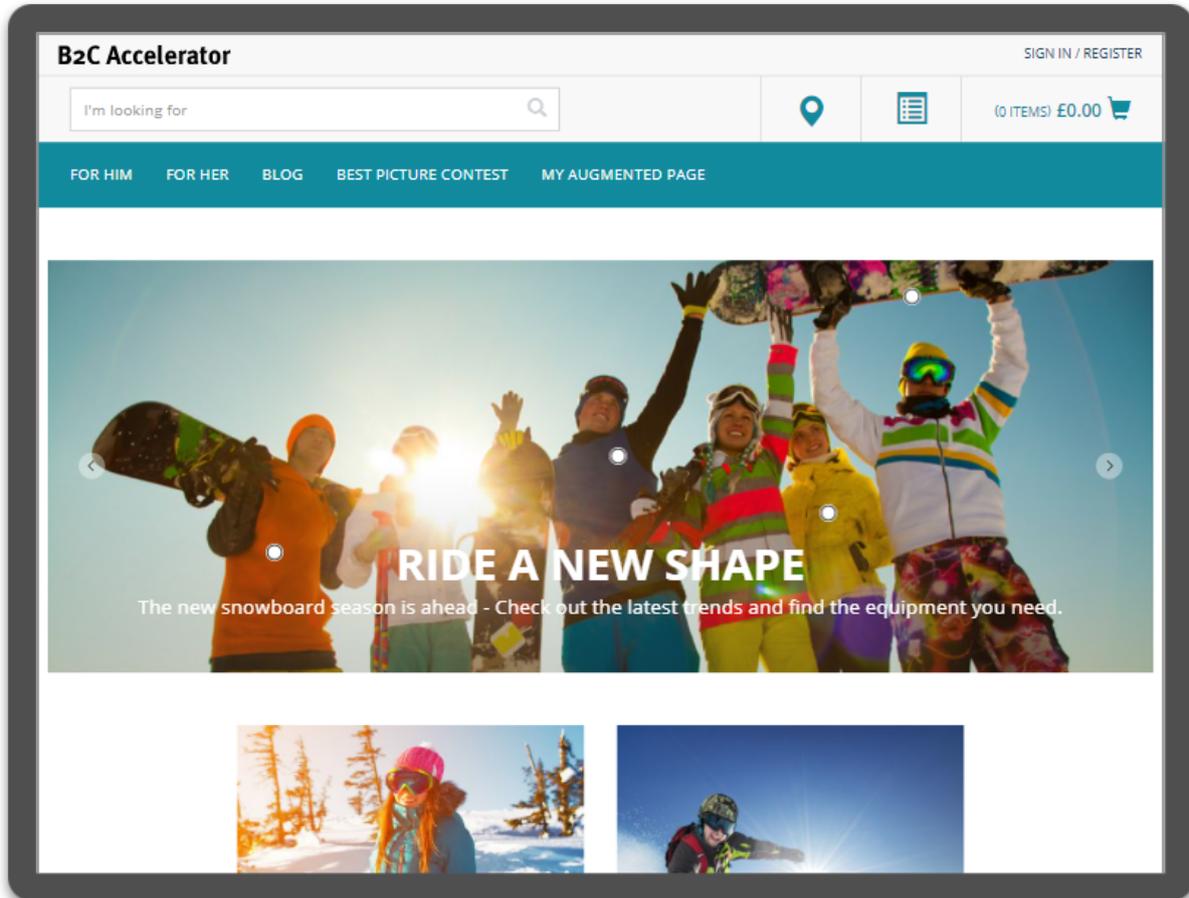
Augment SAP Commerce Cloud storefront
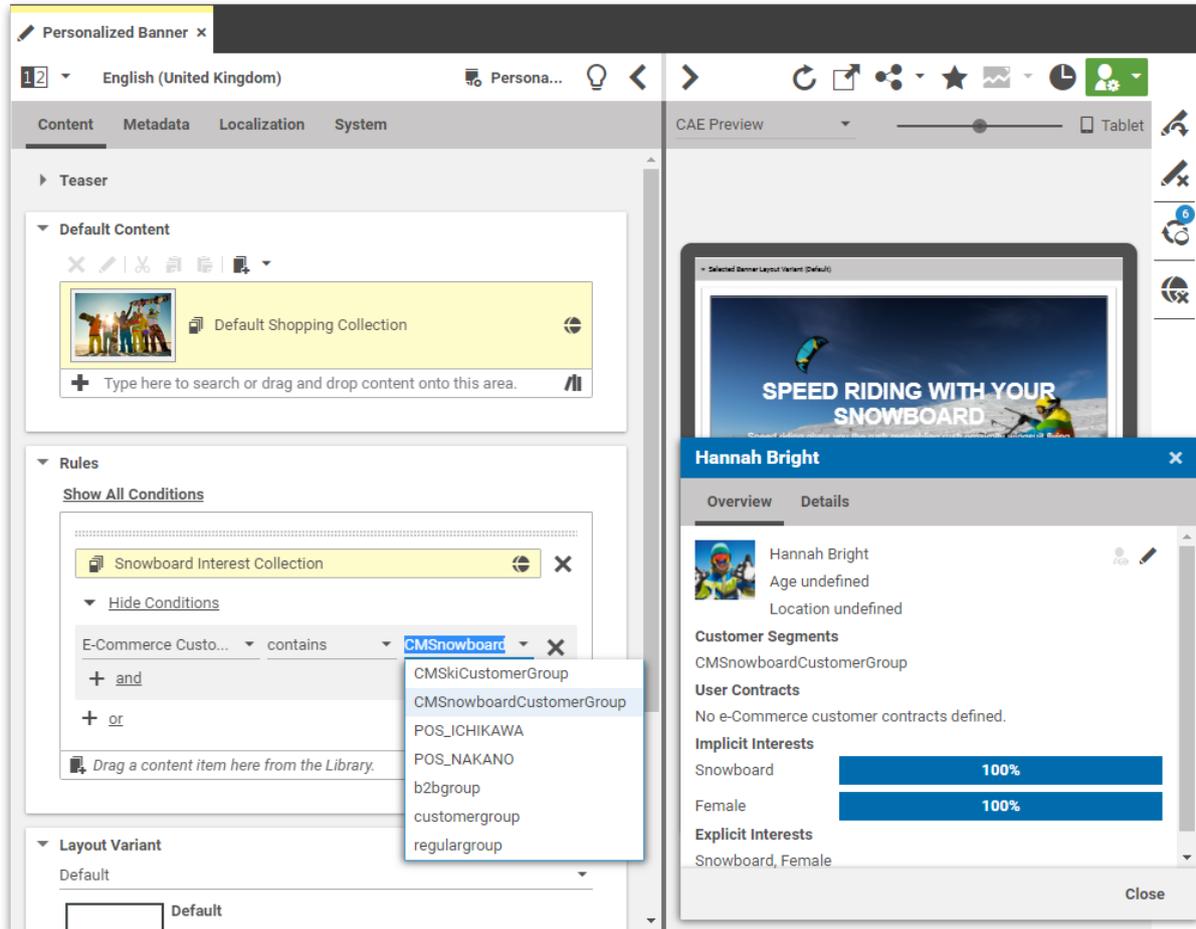
Creating personalization rules based on e-Commerce customer segments

SAP Commerce customers can now upgrade to the latest version of SAP Commerce Cloud to profit from recent enhancements and fixes – as well as to stay in sync within the SAP support lifecycle.

2. Usability and Performance Enhancements

2.1. Headless Server personalization and other optimizations

Headless content management delivers a big advantage for developers by providing clean APIs and easy access to content for all sorts of native app, browser-based Single Page Apps or Progressive Webapps. Until recently, however, it hasn't been as empowering for marketers. CoreMedia's Headless Server was designed to make it easy for developers to create platform agnostic apps, while also giving marketers and merchandisers fine grained control over all of their products and brand assets.

CoreMedia is committing to continuously improving the CoreMedia Headless Server based on feedback from the field, partners, and customers. This latest iteration is focused primarily on providing personalization rules via GraphQL. But it also includes numerous additional performance enhancements.

Based on these new enhancements, editorial users and frontend developers can:

> formulate GraphQL queries for personalized content to be utilized in clients using the headless API. CoreMedia document types like personalized content and user segments are taken into account
> retrieve the folder a content object is located in via repositoryPath in GraphQL queries
> take advantage of the latest version of GraphQL-Java (upgraded to version 15)

In addition to these personalization capabilities, the following improvements will also be available for the CoreMedia Headless Server starting with the 2010.1 release:

> All properties of personalization doctypes have been made available via GraphQL.
> Interfaces can now implement interfaces
> Custom Scalar specification URL
> Improved data fetchers
> Improvements for subscriptions
> Ability to add descriptions to schema
> Support for directives at variable definitions
> All enhancement available as part of GraphQL-Java v15[1]

With this enhanced Headless Server, CoreMedia customers can now leverage personalization use cases in headless scenarios. Rules created by editors can be processed by customer clients and 3rd party personalization engines.

---

[1] https://www.graphql-java.com/documentation/v15/

2.2. Sync Content UUIDs across different environments

A universally unique identifier (UUID) is a 128-bit number used to identify information in computer systems. CoreMedia first introduced UUIDs in distribution 2004.1 as a way to track individual content items in content management servers. In distribution 2007.1, this was extended to uniquely identify editorial comments.

This enhanced feature enables administrator and developers to sync the UUIDs of individual content items between separate computing environments. Not only does this make it easier to track the movement of content across environments, it also allows customers to easily transfer editorial comments between environments.

New capabilities include:

> `serverimport` and `serverexport` have been extended to allow exporting and importing UUIDs of content
> `dumpusers` and `restoreusers` have been extended to allow exporting and importing UUIDs of users
> a new tool that enables a low-level initial sync of UUIDs between environments

```
<SimpleAll folder="/" name="Some Document" uuid="b6854f73-fd2b-4c82-8be8-5e2fbc81b621" ...>
```

UUID field example

This feature enables customers to keep UUIDs consistent between environments and allows for new customization and extension scenarios that requires centralized data storage or that require transfer of custom data between environments.

2.3. Override default usernames with meaningful display names and emails

External user management systems often generate usernames that are not meaningful and do not correspond with a user actual name and email. Meaningful usernames however are important for features like workflows, editorial comments or editing history. Without meaningful usernames, it can be difficult or even impossible for users to know who they are interacting with – such the name of an editor who wrote a comment or the person who started a workflow.

To address this problem, the latest version of CoreMedia Content Cloud (2010.1) enables administrators to attach a meaningful display name and user email to any individual in the user manager.

With this new feature:

- Administrators can assign a new display name and the user email in the user management system of the CoreMedia Studio.
- If available, the new username will be displayed instead of the default username in all relevant scenarios

The new properties (meaningful display name and e-mail) will be recorded in a settings document that is stored in the user's home folder but will be hidden from the user. The information is only available in CoreMedia Studio and will not be written back to the external user management system

Usage of display name in content editing history

Example of a display user with a name in the User Management application
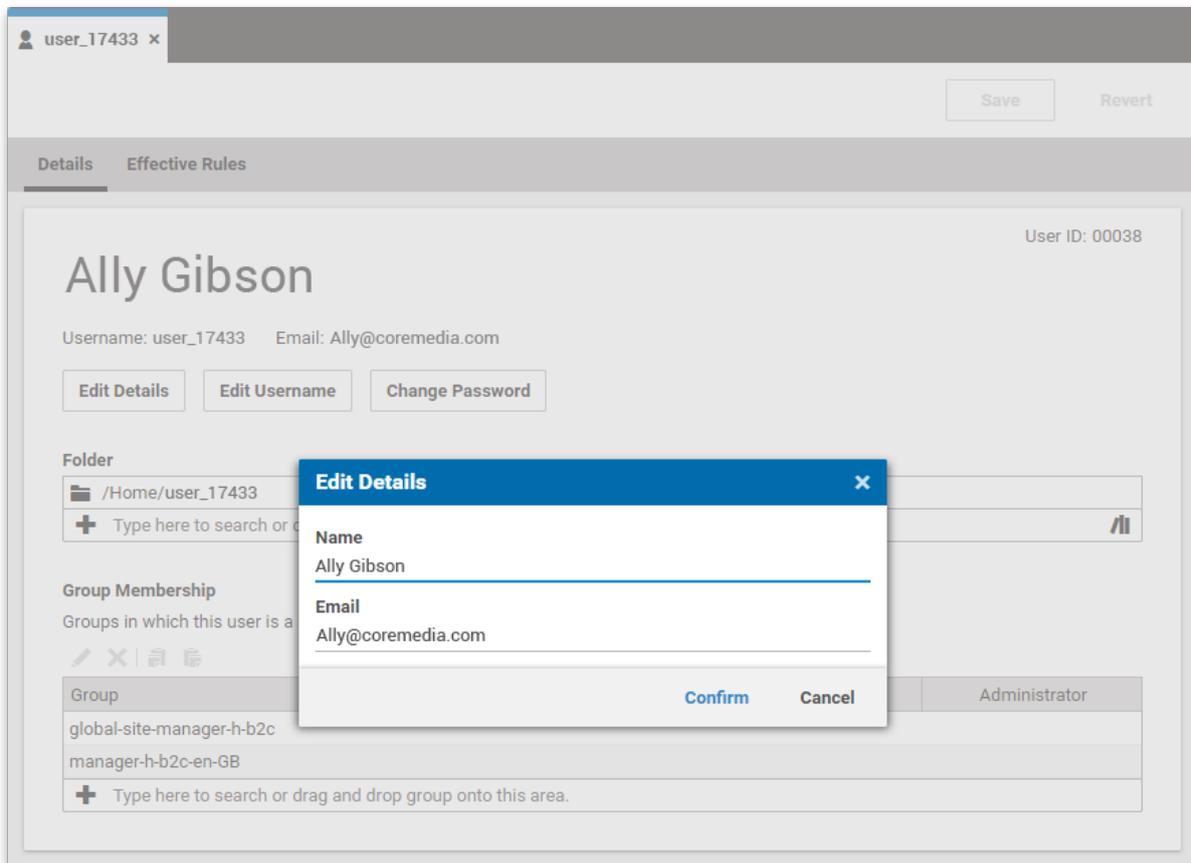
Managing name and email for built-in user management

2.4. Improved tag selection dialog

In previous version of CoreMedia Content Cloud, user interacting with large tag libraries could experience delays when loading many taxonomies or locations in the tag selection dialog of the content metadata tab. In addition, the system did not provide an easy way to search or browse tag existing tag libraries.

In distribution 2010.1, CoreMedia has significantly improved the performance of the tag selection dialog and added new tools to search and browse tag libraries.

Key new features include:

> performance of the tag selection dialog has been improved.
> A search field and tag browser have been added to help users locate specific tags and improve the handling of tags and taxonomies
> The new search & browse functionality includes incremental, real-time search suggestions and bested taxonomies
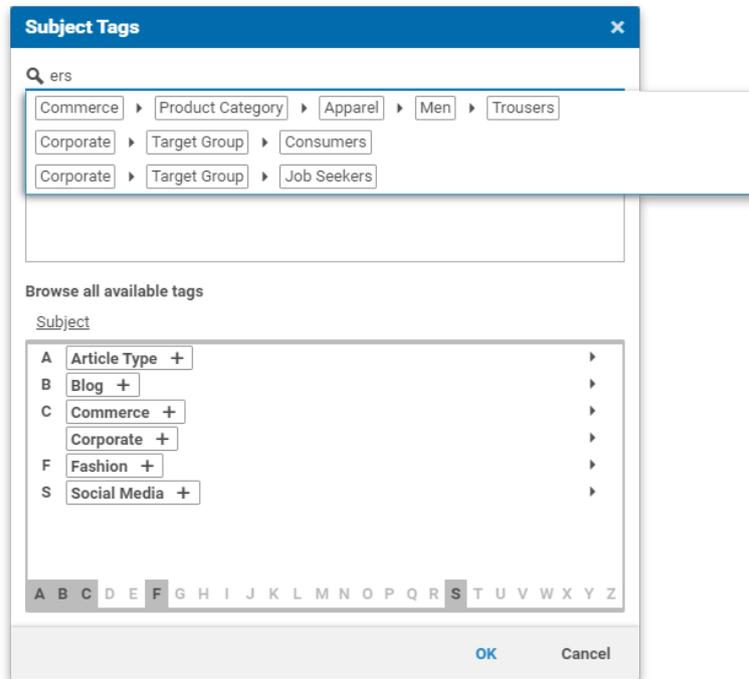
Search and browse interface in enhanced tag selection dialog

Adding taxonomies and locations to content items is now more efficient due to these performance improvements. Users can locate and apply tags more efficiently and gain a better understanding of the full scope of available tags and taxonomies.

2.5. AWS S3 now available as Contentserver media storage

CoreMedia customer have requested a way to store media files in the form of binary large objects (BLOBs) in an external storage system that is independent of any specific data center or region.

CoreMedia has addressed this need by extending the Contentserver so that it customers can utilize Amazon Web Services (AWS) S3 cloud storage as a BLOB store. This makes it easier to use the CoreMedia CMS in an AWS cloud environment. The advantage of this approach over other storage options is that AWS provides extremely high reliability at low costs, while maintaining independence form from availability zone (i.e. datacenter) of a particular AWS region.

Access to the AWS S3 Blobstore was first introduced in distribution 1904 as part of the CoreMedia Content Cloud Hosted Service. In 2007 the API was enhanced so that it is now ready to be used by CoreMedia Content Cloud Self-Managed customers.

This feature is now a standard feature of the CoreMedia Contentserver (CS, MLS, RLS). The feature works out-of-the-box but is not the default setting and must be configured by the customer like any other external blob store).

This feature provides all CoreMedia customer (including Hosted and Self-Managed) with the ability to store BLOBs outside the database (e.g. in a file system on a local disk or local network). In a cloud environment, such storage is usually bound to a datacenter. However, the Contentserver should be deployable in any datacenter in any region, which would make it necessary to copy BLOBs between datacenters. This feature makes it possible to store BLOBs in a storage service that is independent of any specific datacenter in given region. Overall, this feature enhances cloud readiness of the CoreMedia CMS.
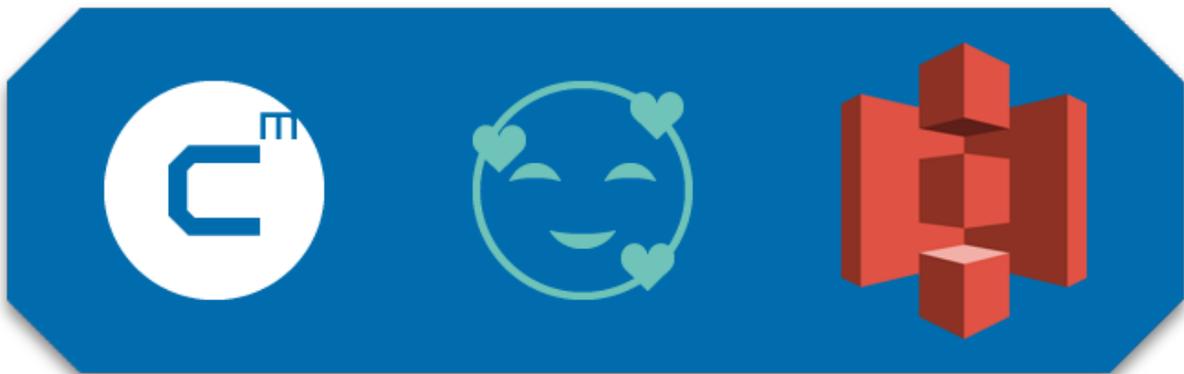
2.6. HTTP/2 now supported in Docker Deployments

HTTP/2 is a major revision of the HTTP network protocol used by the World Wide Web. It is the first upgrade to the Hypertext Transfer Protocol since 1999.

HTTP/2 introduces several new features designed to improve page load times for website visitors, including:

- Multiplexing: allows a browser to include multiple requests in a single TCP connection
- Header compression: forces all HTTP headers to be sent in a compressed format, reducing the amount of information that needs to be exchanged between browser and server.
- Server push: eliminates unnecessary round trips and speeds up page load times by sending web assets (e.g. CSS stylesheets and image resources) back to the browser before it knows it needs them.
- Stream priority: a mechanism for browsers to specify which assets they would like to receive first (e.g. page first, followed by CSS, JavaScript, and image assets).

CoreMedia has been using HTTP/2 as the standard in all of its core applications for some time. With distribution 2010, CoreMedia has enabled it for Docker deployments.

All components in CoreMedia CMS which process web requests now support HTTP/2. By updating all underlying components like servlet containers and web frameworks in older releases, CoreMedia has now ensured that all web servers are configured to use HTTP/2 by default for Docker deployments.

Benefits of this approach include[2]:

> Low overhead in parsing data — a critical value proposition in HTTP/2 vs HTTP1.
> Less prone to errors.
> Lighter network footprint.
> Effective network resource utilization.
> Eliminating security concerns associated with the textual nature of HTTP1.x such as response splitting attacks.
> Enables other capabilities of the HTTP/2 including compression, multiplexing, prioritization, flow control and effective handling of TLS.
> Compact representation of commands for easier processing and implementation.
> Efficient and robust in terms of processing of data between client and server.
> Reduced network latency and improved throughput.

---

[2] Source: https://medium.com/@factoryhr/http-2-the-difference-between-http-1-1-benefits-and-how-to-use-it-38094fa0e95b

3. New CoreMedia Labs Features

3.1. Spark – The React Example App
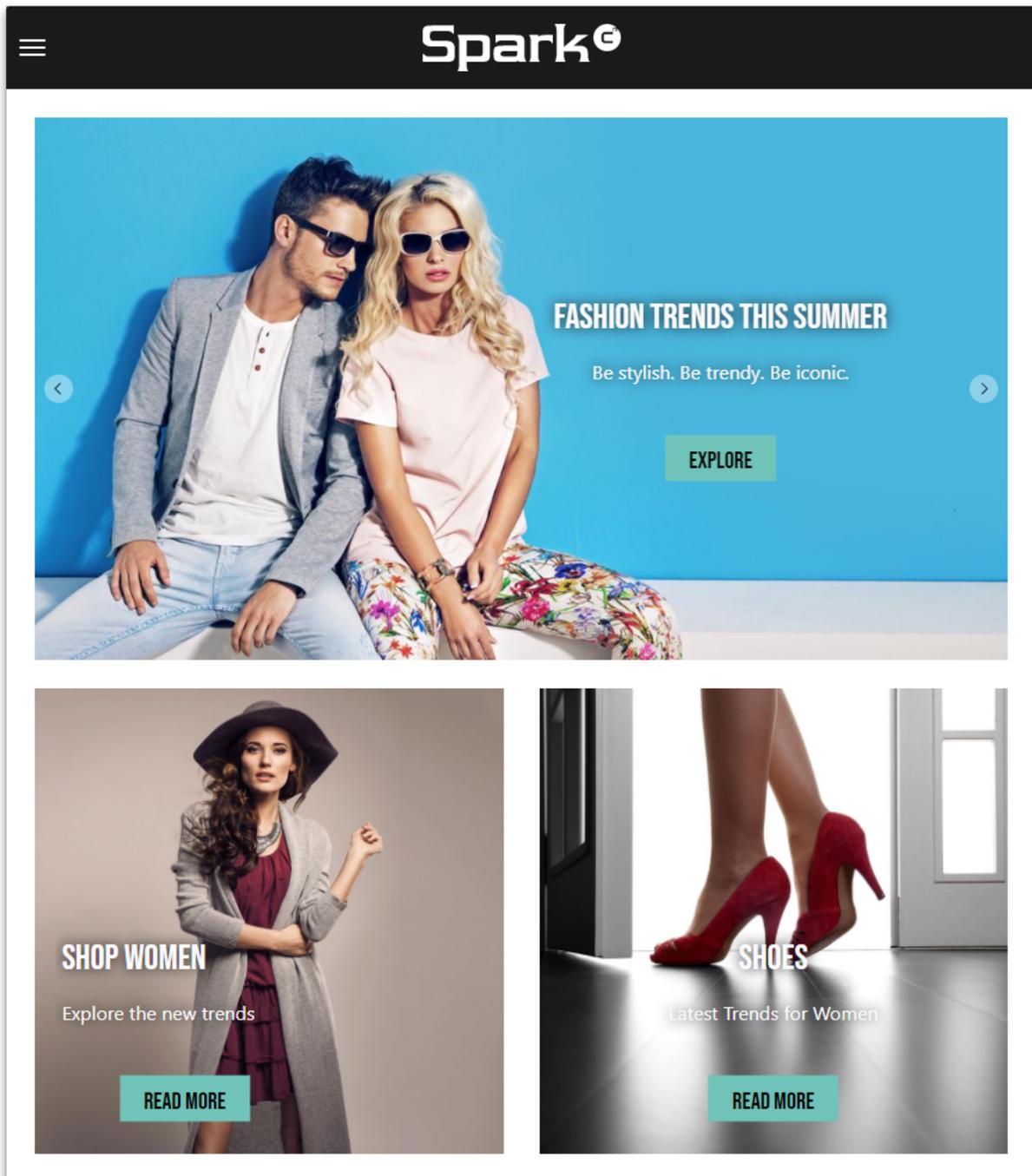
Spark, our new example React application

Due to the increasingly popularity or headless content and commerce systems, developers are often faced with the requirements to build custom user interfaces for single-page applications (SPAs) or progressive web apps (PWAs). In order to reduce development time, increase performance, and allow its customers to leverage best practices, CoreMedia has developed a standard, React-based SPA starter site called Spark.

React is an open-source, front end, JavaScript library for building user interfaces or UI components. It is maintained by Facebook and a community of individual developers and companies. React can be used as a base in the development of single-page or mobile applications. (Source: Wikipedia)

Developers often need some guidance and examples to know how to best develop applications by means of the data provided by the CoreMedia Content Cloud and its Headless server. CoreMedia customers can use Spark as a quick start for building prototype applications. Note that we do not recommend the use of these prototype apps in production environments.

This new feature from CoreMedia Labs includes the following:

> A simple React application to provide guidance for frontend developers that are building websites and other applications based on the CoreMedia Headless server.
> Code examples and documentation.

Although we chose React with which to build Spark, the principles can easily be transferred to other component libraries and frameworks such as Vue and Angular.

The Spark React application and its accompanying guide will be published on CoreMedia Labs soon after the 2010 AEP release. The 2010 AEP is the prerequisite for this application.

CoreMedia provides the perfect foundation for front end developers to build and deploy React-based, Vue, or Angular apps to support any headless content or commerce environment.

3.2. New Content Hub Connectors

The CoreMedia Content Hub is a powerful tool for the editorial process. It allows editors to integrate with any kind of 3rd-Party data platform. From classic DAM systems, to media platforms such as YouTube, to file-sharing Platforms such as Dropbox or even web feeds such as RSS. Editors can then browse through the content from within the CoreMedia Studio Library and import content as needed into the CMS. Content-Connectors to YouTube and RSS channels are available out of the box.

The Content Hub officially replaced the earlier Studio Hub Labs project and initially shipped with two example adapters for integrating RSS feeds and YouTube.

In response to customer requests and to increase editorial efficiency, the Content Hub has been extended with the following new connectors to access content directly from the CoreMedia Studio:

> Cloudinary Connector: Allows editors to access assets from the Cloudinary image and video management system
> Dropbox Connector: Allows editors to access files from a Dropbox shared file system
> Filesystem Connector: Allows editors to directly access content on local or networked filesystems
> CoreMedia Connector: Allows users in testing and staging environments to access content stored in a CoreMedia production repository
> CoreMedia Navigation Connector: Displays the contents of the customer's repository based on the navigational structure of the relevant web site or online store
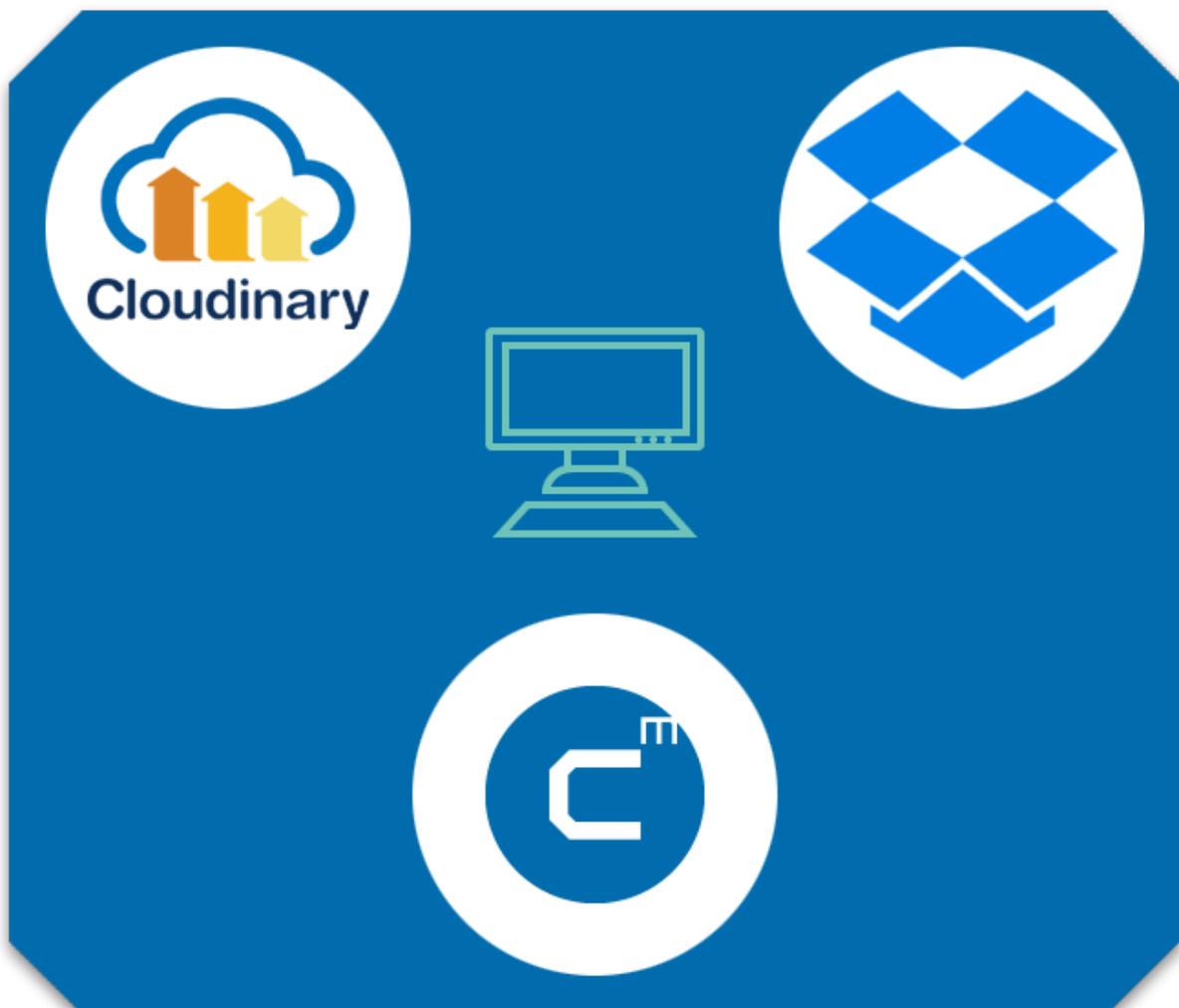
With these new connectors, editors can directly integrate Dropbox, the file system, Cloudinary, and other CoreMedia instances into the CoreMedia Studio. Users can browse these repositories and import them into their content without leaving the CoreMedia Studio.

These five new connectors will be published on CoreMedia Labs soon after the 2010 AEP release. The 2010 AEP is the prerequisite for this application. They will be released as source code packages on GitHub and must be compiled and integrated as extensions into the customer's Blueprint workspace.