Magnolia Headless Content & Salesforce Commerce Cloud

Version 20.1.0

magnolia®

salesforce

commerce cloud

Contents

Summary

Amplify your content commerce experience and increase sales by combining the power of Magnolia with the Salesforce Commerce Cloud (SFCC) APIs and Composable Storefront. Whether you are already using Magnolia or have recently decided to 'up your customer experience game', this integration will effectively and quickly connect these three systems.

Editors and digital retailers can create engaging content beyond simple store fronts - including compelling calls to action with direct links to product detail pages. With Magnolia, editors can centrally manage content and roll it out globally to all markets and in multiple languages. They can also plan and create campaigns with WYSIWYG and preview capabilities.

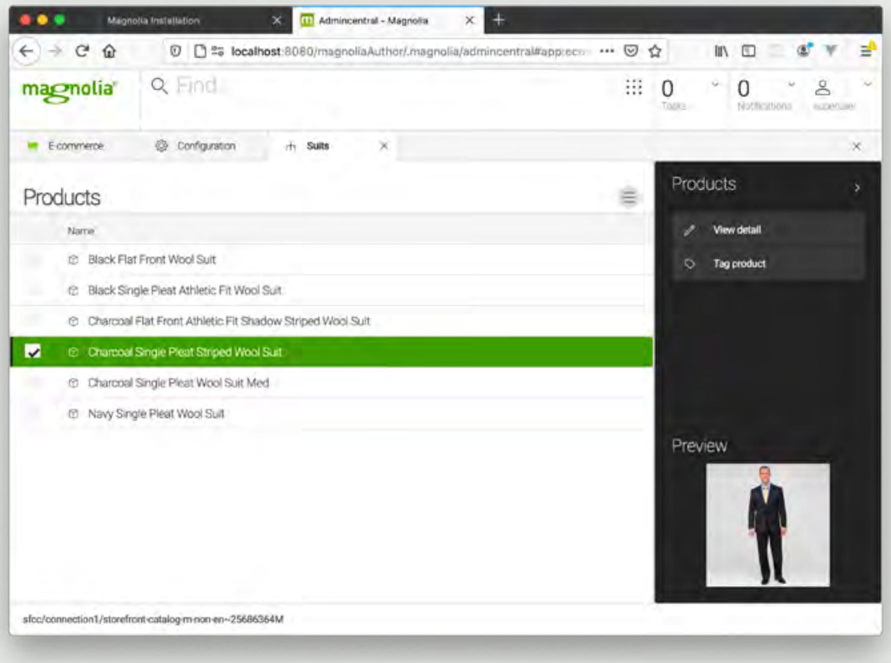
Editors will be comfortable and in control, working in the light, modern, user-friendly Magnolia interface.

The Magnolia + SFCC APIs and Composable Storefront integration empowers your users and increases your sales.

A Magnolia DX Core license and installation is required to use this integration. Magnolia can be deployed in an on-premise or cloud configuration. The license is to be purchased directly through the Magnolia sales department.

This integration is installed in your Magnolia instance. It connects to your SFCC instance per REST API to provide 5 features.

Your Storefront frontend easily integrates with Magnolia via Magnolia's rich content and commerce REST APIs and React components.



Name
Black Flat Front Wool Suit
Black Single Pleat Athletic Fit Wool Suit
Charcoal Flat Front Athletic Fit Shadow Striped Wool Suit
☑ Charcoal Single Pleat Striped Wool Suit
Charcoal Single Pleat Wool Suit Med
Navy Single Pleat Wool Suit

sfcc/connection1/storefront-catalog-m-non-en~25686364M

SFCC products displayed in the Magnolia authoring interface

Integration overview

Functional Overview

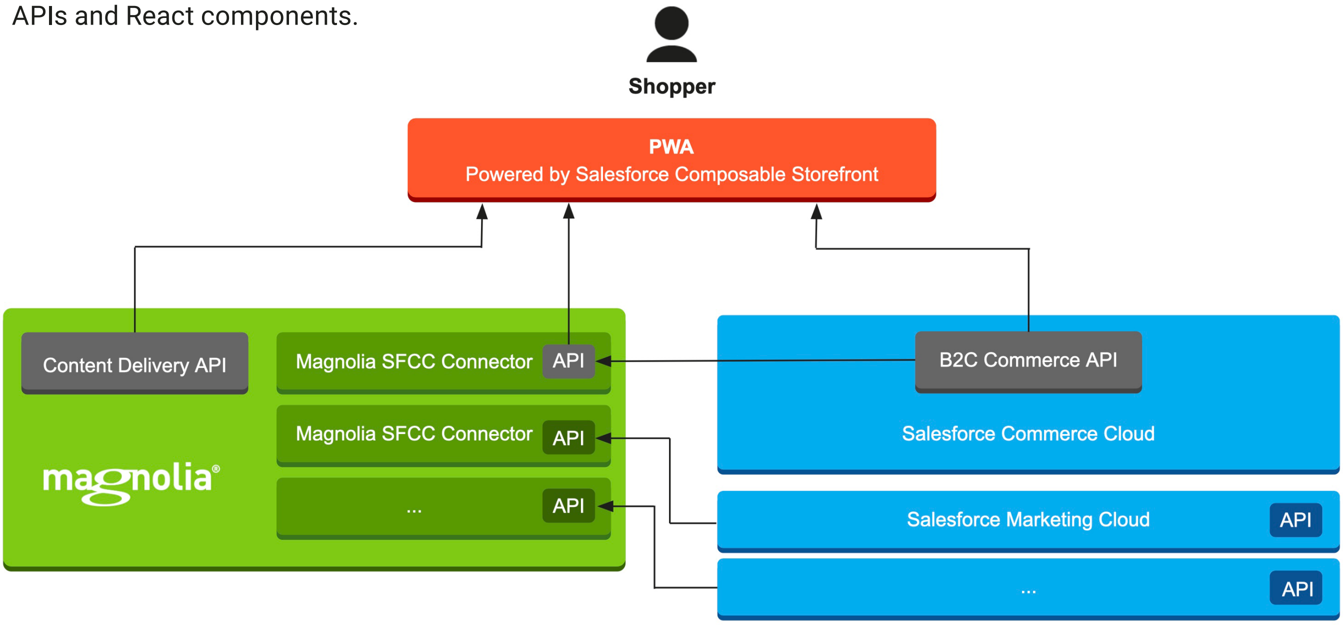
This integration is installed in your Magnolia instance, connecting it to your Salesforce Commerce Cloud (SFCC) instance in 5 ways.

1. Your Magnolia interface (AdminCentral) will display your SFCC products and catalogs.
2. Magnolia users can browse and choose catalogs and products and link them to Magnolia content items (or even to content items from other connected systems, such as a DAM).
3. Apply tags to SFCC products within Magnolia based on A.I. powered text and image recognition. Tags can be used for search and for dynamic product list generation, such as personalization. (Tags are stored within the Magnolia instance.)
4. Magnolia's global search tool will include SFCC catalogs and products in addition to Magnolia content.
5. Magnolia's content APIs return SFCC catalogs and products for use in headless implementations.

Your Storefront frontend easily integrates with Magnolia via its rich content and commerce REST APIs and React components.

1. The Storefront frontend retrieves text and media content from Magnolia and displays it.
2. Retrieved content will include references to SFCC products and catalogs. (These references were created by content authors in Magnolia with its 'link' feature.)
3. The Storefront frontend retrieves the desired information from SFCC based on the links with the OCAPI or Commerce APIs, and displays it.
4. The Storefront frontend displays cart and checkout features by using either SFCC cart APIs.

The integration is implemented using a software module installed on the Magnolia instance, the Magnolia SFCC connector. There is nothing installed in the SFCC instance. A client_id or token must be generated in the SFCC system and provisioned in the Magnolia instance. The Magnolia SFCC connector module communicates with SFCC via the OCAPI REST API.



Use Cases

Members of the marketing team, digital content team, digital retailers, merchandisers, media booking team and other business users (collectively referred to as 'authors') use the Magnolia CMS web-based application to create and manage content and commerce experiences and campaigns. Thanks to the integration, SFCC products and catalogs are available directly in the Magnolia UI. The shopping experiences the authors create are delivered via a performant modern web app thanks to the Composable Storefront.

With Magnolia, authors have access to enterprise-class content use cases:

- Editors can pick and place products straight from the SFCC catalog to create content with true call to action capabilities.
- Create landing pages (Full WYSIWYG and multi device preview).
- Targeted content rollout to multiple sites based on global multilingual content.
- Magnolia allows you to deliver content in all locales and languages.
- Create and deliver slot-based campaigns .
- Create and deliver fragments for product categories.
- Create and deliver content for brochures and magazines.
- Asset management with focal points and image hotspots. Image hotspots can directly link to SFCC product detail pages.
- Pick and place forms from external systems like Salesforce Marketing Automation (available in a separate integration).
- Check content performance by pulling in analytics data from external systems straight to the content.

See the 'User Guide' section for details.

Limitations, Constraints

No limitations. You can use all the functionality of Magnolia CMS. There are no limitations with respect to the SFCC APIs and Storefront.

Compatibility

The Magnolia SFCC connector module communicates with SFCC via the OCAPI REST API.

The Storefront frontend may communicate with SFCC via the OCAPI or SF Commerce API.

The Storefront frontend becomes editable due to the Magnolia react-editor npm package.

Tested against OCAPI 20.4.
Tested against Storefront 2.4.3 (@mobify/create-app & @mobify/commerce-integrations)
Tested against Magnolia React-Editor 1.0.6 (@magnolia/react-editor)

The Magnolia SFCC connector module exists for all 6+ versions of Magnolia.

Magnolia SFCC Connector Module version	Magnolia CMS version
1.2.3	**6.2.5+**
1.2.2	6.2.5
1.2.1	6.2.3
1.2	6.2.2
1.1.1	6.2.2
1.1	6.2
1.0	6.1

Privacy, Payment

This integration does not access customer profile data, and performs no credit card activity itself.

This integration connects to the SFCC OCAPI API, which could be used to trigger a checkout.

→ Magnolia Commerce APIs

→ Magnolia Content Delivery API

→ Magnolia Security Best Practices for developers

→ Magnolia Security Policy

Implementation Guide

Architectural Considerations

Headless Dependencies

The headless approach enables each subsystem to be developed and managed independently to a high degree. This makes the overall architecture very flexible, and allows the teams working on each subsystem more freedom in how their component is implemented. The 'headless' approach can be seen as a type of Service Oriented Architecture in the content and commerce space. The key dependencies are the APIs of each system. Many problems are avoided by versioning of APIs, which enable dependents to upgrade/migrate to new versions at their convenience.

Headless Frontend (Salesforce Composable Storefront)

Thanks to the headless approach, frontends can now be generated in the technology of choice. Frontend frameworks such as React are a good choice as they are popular and well known, making it easy to find developer talent and allowing developers to get up to speed quickly. They provide an excellent developer experience and enable great app-like customer experiences as well. The frontend can be delivered as a Single Page Application (SPA) or a variant thereof such as a Progressive Web App (PWA).

Server for Frontend (Salesforce Composable Storefront)

However, a purely browser based frontend (i.e. SPA/PWA) has some shortcomings. It can be difficult to make secure API calls directly from the browser to the Ecommerce, CMS, or other services. Search engine rankings can be lowered (SEO). The performance on the first visit to the site can be poor. Also, social previews, such as in tweets and Instagram posts, may miss the proper text and images for the site. All of these problems are solved by a serving solution including server side rendering (SSG) such as Salesforce Composable Storefront.

Magnolia, SFCC APIs & Storefront

The integration is implemented by way of a software module installed on the Magnolia instance, the Magnolia SFCC connector. There is nothing installed in the SFCC instance. Only a client_id or token must be generated in the SFCC instance so that Magnolia can securely access the commerce server.

Retailers manage products in SFCC Business Manager.

Authors use Magnolia to create content.
They can browse the SFCC catalog and products, achieved via SFCC OCAPI APIs.
They can add a product to a Magnolia content item, this causes a reference (ie ID) of the product to be saved in the Magnolia content.
In the pages app - authors interact with a WYSIWYG visual preview of the Storefront-based Frontend, served directly from the Salesforce Managed Runtime over standard HTTP requests.

Visitors access the Storefront Frontend.
The Storefront loads the content from Magnolia via the Content Delivery API. In the response, it finds all of the references to SFCC items, retrieves those items via SFCC API's, and then renders the HTML and returns this to the browser.
The Storefront uses SFCC OCAPI API to provide Cart functions.

Magnolia WYSIWYG Editing and Preview

In addition to the standard headless CMS approach, the Magnolia Pages App provides a powerful, fully headless authoring approach where authors can see their site, and choose and place available frontend components to create experiences visually. This approach is sometimes referred to as 'hybrid headless'.

Frontend setup:

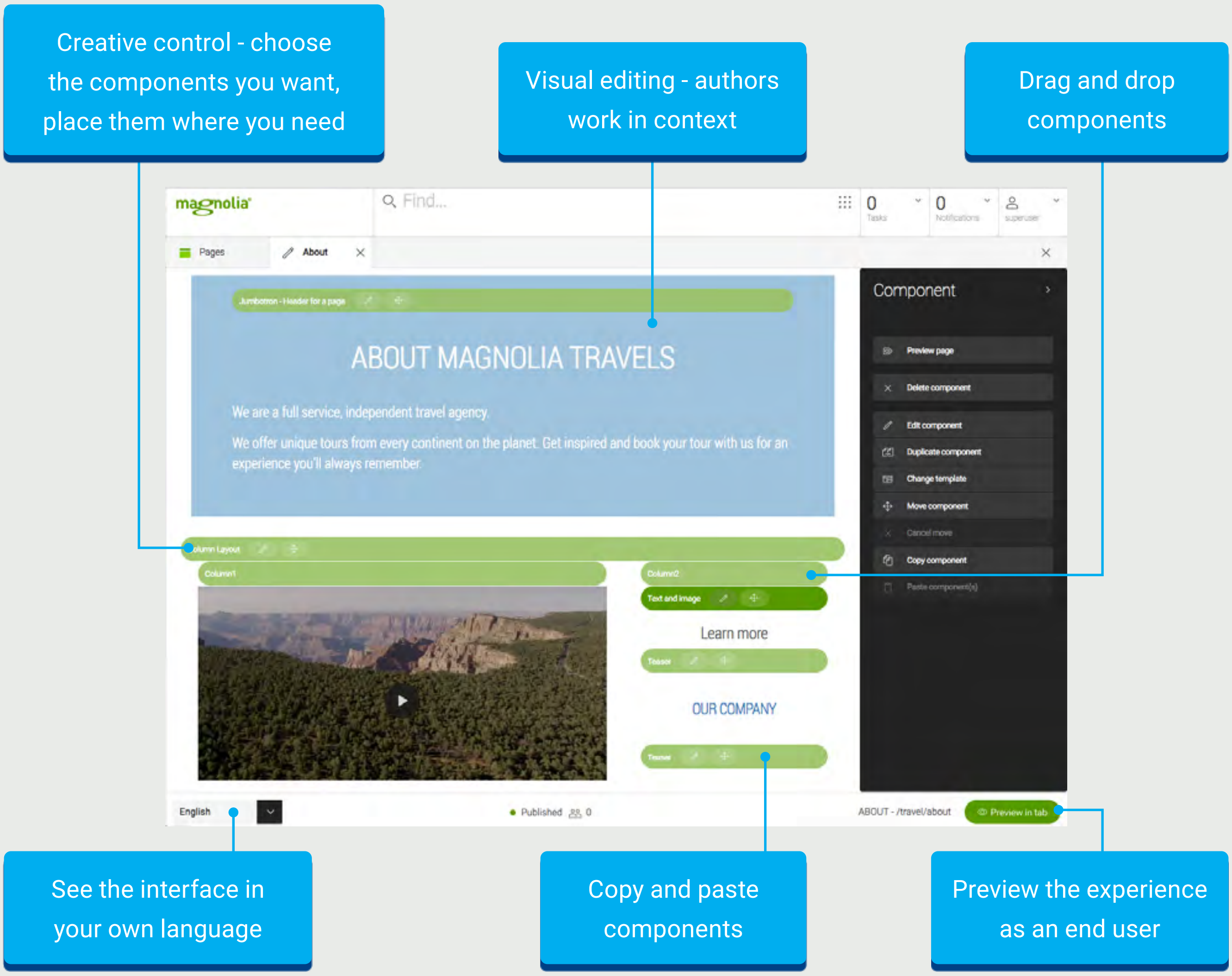
- Frontend developers add a small Magnolia library to the frontend project. Libraries support React, Vue and Angular. React is used to support Storefront.
- The frontend hits the Magnolia Content API to retrieve a 'page'.
- The page response contains a 'tree' of content. Each content node specifies which frontend component it is.
- The frontend instantiates the React (or other) component specified.

Authoring setup:

- Developers configure a form for each component, specifying which properties an author can add. (Similar to a content type.)
- Developers configure which components can be added to which areas of a page, or to which areas of other components. (Useful for carousels, responsive layout rows, etc)
- The Page editor loads the actual frontend in an IFRAME and is able to provide editing UI based on the configuration. After every change the frontend is re-rendered and displays the changes.

→ Full documentation and tutorial

Hybrid Headless: full experience control



Creative control - choose the components you want, place them where you need

Visual editing - authors work in context

Drag and drop components

See the interface in your own language

Copy and paste components

Preview the experience as an end user

ABOUT MAGNOLIA TRAVELS

We are a full service, independent travel agency.

We offer unique tours from every continent on the planet. Get inspired and book your tour with us for an experience you'll always remember.

Setup of Business Manager

N/A - Nothing needs to be done in Business Manager.

Configuration

An API client must be created in SFCC Account Manager. This is required so that Magnolia can request content from the SFCC servers. No other configuration is required in the Business Manager or Account Manager.

Configuration of Magnolia

Installation and Configuration is required in Magnolia.

REST Endpoints must be configured in Magnolia to enable access to the content to the Frontend. Content can be made publicly available or be restricted or partially restricted.

The SFCC Connector is a part of the Magnolia Commerce Connector Pack.

Several values from SFCC must be configured in Magnolia so that it can request content from SFCC.

Complete documentation of the Commerce Connector Pack (including sub pages)

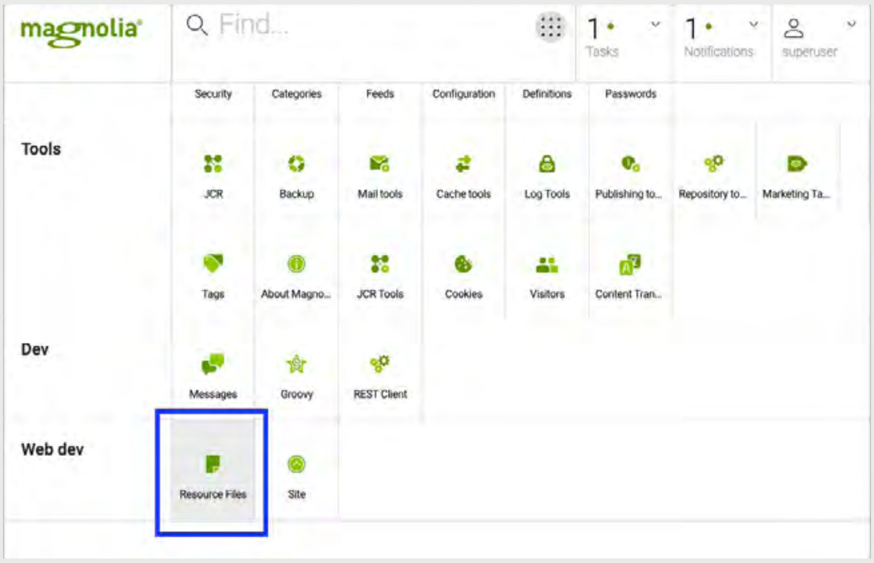
Installation and Configuration

Quick start

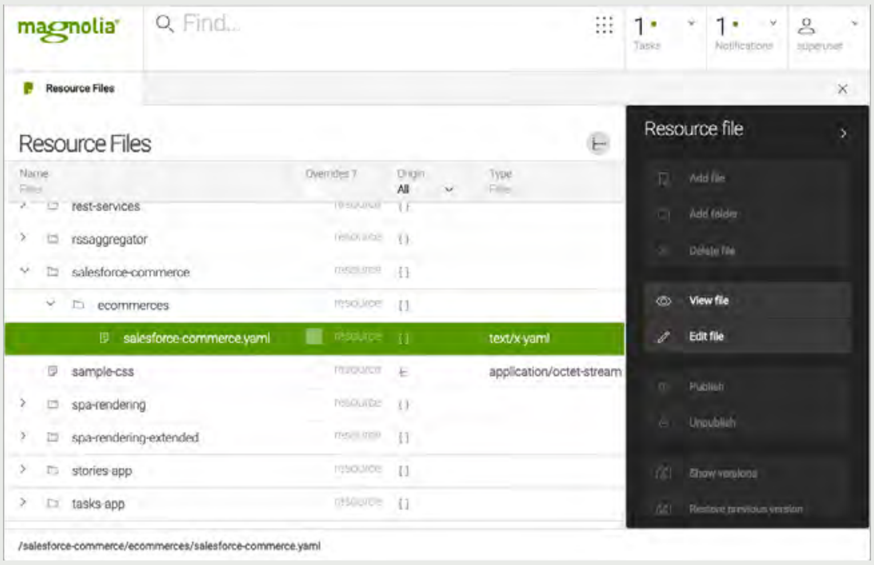
There are several ways to configure items in Magnolia. Here is a basic approach in case you are not familiar with Magnolia.

Open the **'Resources files'** app.

Scroll down to **'salesforce-commerce'** and expand the nodes.



Select salesforce-commerce.yaml. Click 'Edit file'.

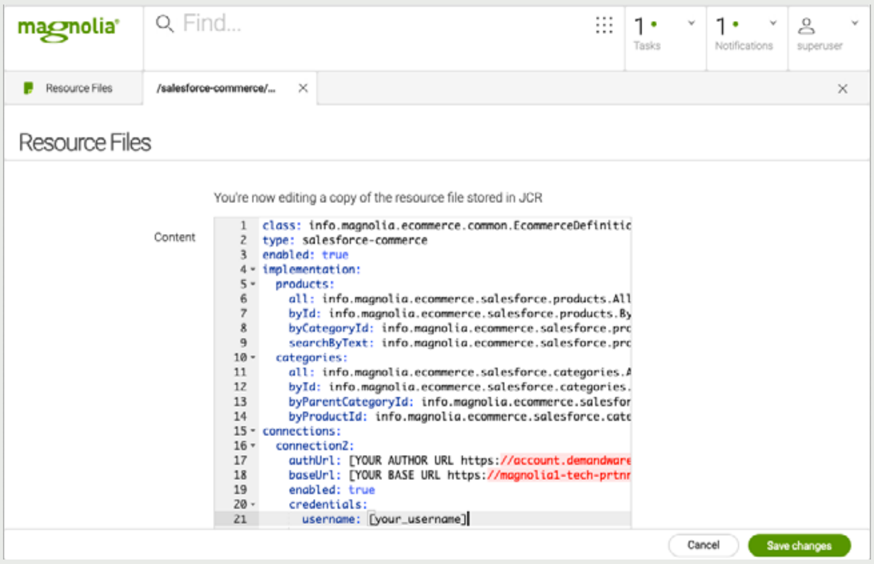


Resource Files

Name	Overrides?	Origin	Type
rest-services		resource	
rssaggregator		resource	
salesforce-commerce		resource	
ecommerces		resource	
salesforce-commerce.yaml		resource	text/x-yaml
sample-css		resource	application/octet-stream
spa-rendering		resource	
spa-rendering-extended		resource	
stories-app		resource	
tasks-app		resource	

/salesforce-commerce/ecommerces/salesforce-commerce.yaml

Paste in your configuration (overwriting the existing configuration) and save.



Resource Files

You're now editing a copy of the resource file stored in JCR

```
class: info.magnolia.ecommerce.common.EcommerceDefinitic
type: salesforce-commerce
enabled: true
implementation:
  products:
    all: info.magnolia.ecommerce.salesforce.products.All
    byId: info.magnolia.ecommerce.salesforce.products.By
    byCategoryId: info.magnolia.ecommerce.salesforce.prc
    searchByText: info.magnolia.ecommerce.salesforce.prc
  categories:
    all: info.magnolia.ecommerce.salesforce.categories.A
    byId: info.magnolia.ecommerce.salesforce.categories.
    byParentCategoryId: info.magnolia.ecommerce.salesfor
    byProductId: info.magnolia.ecommerce.salesforce.cate
connections:
  connection2:
    authUrl: [YOUR AUTHOR URL https://account.demandware
    baseUrl: [YOUR BASE URL https://magnolia1-tech-prtnr
    enabled: true
    credentials:
      username: [your_username]
```

Please remember that the Salesforce credentials must be always stored in the Magnolia password manager for security reasons. Please have a look at the clientSecret configuration as documented here.

External Interfaces

The Magnolia SFCC API Endpoints are documented here: https://documentation.magnolia-cms.com/display/E-commerce+REST+endpoints

Configuration of the Frontend

The Storefront-based frontend is customized and developed to meet the needs of the project.

The URLs and any necessary security credentials to access the Magnolia content API's will need to be configured.
The URLs and any necessary security credentials to access the SFCC API's will need to be configured.

Storefront Example project: Authoring & Frontend

A simple example of this integration has been created which includes the full authoring experience configuration as well as the Storefront React frontend. It is provided as an example only and is not a sample app or template.

For the authoring experience, look within the 'Storefront-lm' directory, view the contentTypes, apps, dialogs & templates directories.

For the frontend, view the 'Storefront' directory.

https://git.magnolia-cms.com/projects/DEMOS/repos/sfcc-magnolia-headless-demo

Implement the Authoring Experience

Developers have two key responsibilities, configuring the authoring tools and creating the frontend. Both of these activities depend on the content model of the project.

Therefore, the first task is to work with all stakeholders to define the content model for the project. This means:

- What are the content types, what are their properties, and what are their relationships?
- In the special case of the WYSIWYG visual page editor -
    - What components are available, and what properties do they have?
    - What page-types are available?
    - What components can be placed in which areas of the pages or other components?

Developers then implement the authoring experience based on the defined content model.

In Magnolia this is done with light YAML-based configuration files. Since everything is in simple text files, developers can use their editors of choice and get all the advantages of git or other source control (change tracking, branching, visibility, etc.

For the classic headless approach, developers create content types & apps to edit them.
For the WYSIWYG visual page editor, they create dialogs & template definitions.

Developing with Magnolia:
https://hd.magnolia-cms.com

Implement the Frontend Experience

Teams have the flexibility to implement frontends according to their technical preferences. Salesforce Composable Storefront provides a powerful modern platform for React-based frontend development. Magnolia React Editor makes it easy to give authors control in headless PWA projects.

Developing with Storefront
https://dev.mobify.com/v2.x/

For Magnolia specific details:
Magnolia Content Delivery API
https://hd.magnolia-cms.com/docs/api/delivery

Visual SPA Editor (for WYSIWYG visual page editing):
https://hd.magnolia-cms.com/docs/visual-spa-editor

Testing

To test the integration, follow the steps provided in the 'User Guide' section of this document.

Operations, Maintenance

Availability

Magnolia - On-premise deployment:

Availability, fallback, and performance depend on the specifics of the Magnolia On-premise deployment and are the customer's responsibility..

Magnolia - Cloud deployment:

Expected availability is 99.5% / month. If that is not met then there is a service credit available to the customer based on the following:
99.5% --> 0
99.4% --> 10% x (yearly Subscription Fee / 12)
99.3% --> 20% x (yearly Subscription Fee / 12)
99.2% --> 30% x (yearly Subscription Fee / 12)

Cloud comes standard with a redundant production public server.

All Magnolia instances can expose a 'status' REST endpoint which can be pinged for system health:
https://docs.magnolia-cms.com/product-docs/Developing/API/REST-API/REST-services-health-check.html

Each Cloud subscription comes with access to Datadog, where users can view dashboards, metrics, monitoring, and logs

More information: https://docs.magnolia-cms.com/product-docs/Magnolia-Cloud/Managing-environments-using-the-Magnolia-cockpit/Monitoring-and-logging.html

Performance is highly dependent on project implementation. Projects with small content payloads and following performance best-practice can expect:
Time to first byte < 0.5 sec
Time to interact < 1.0 sec

On Cloud, datadog can be configured to meet customer needs when it comes to notifications and alerting.

Failover/Recovery Process

Magnolia - On-premise deployment:

Please contact your IT department or Magnolia technology provider

Magnolia - Cloud deployment:

The Magnolia Cloud Helpdesk ensures 24/7 availability and is your single point of contact for all standard administrative issues such as additional accounts and basic support requests.
https://mgnl.io/cloud-helpdesk

Critical incident phone support (24x7x365)
+420 770 610 177

Support

contact@magnolia-cms.com

Additional resources:

→ Magnolia Cloud Helpdesk

→ Magnolia Documentation

→ Magnolia Community

→ Magnolia Support

For advanced services requests, please contact your Customer Account Manager.

User Guide

Roles, Responsibilities

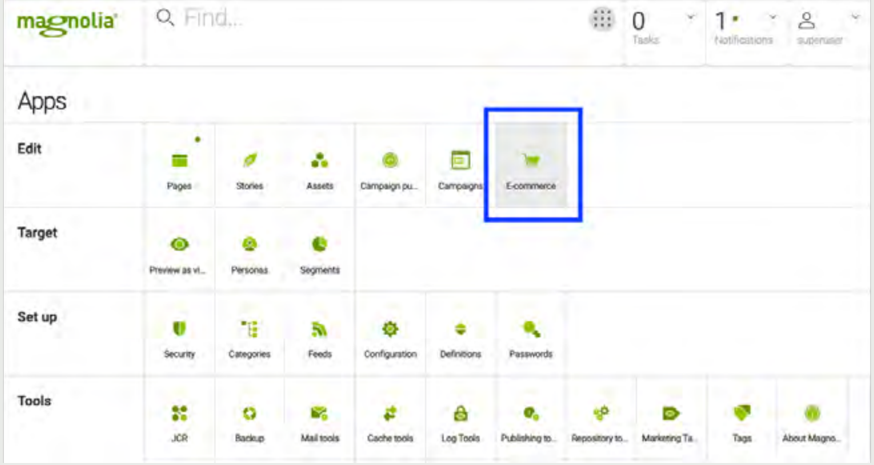
The key role for this integration is the content author. Authors can come from many departments. They may be members of the marketing team, digital content team, digital retailers, merchandisers, media booking team or other business users.

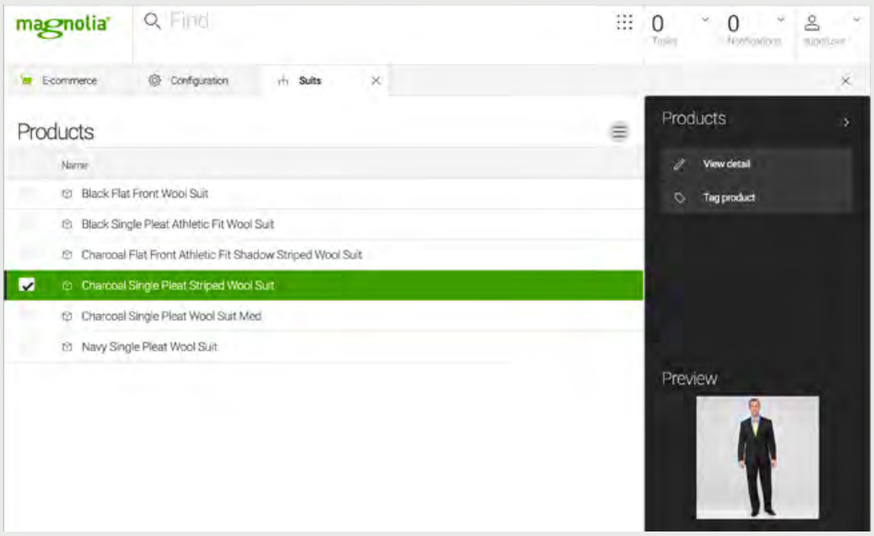
Authors use the Magnolia CMS web-based application to create and manage content and commerce experiences and campaigns. Thanks to the integration, SFCC products and catalogs are available directly in the Magnolia UI. The shopping experiences the authors create are delivered via a performant modern web app thanks to Storefront.

Authors use the Magnolia SFCC Integration in these ways:

View SFCC products and catalogs in the Magnolia E-Commerce App

Authors can browse all configured catalogs in a purpose-built app within Magnolia



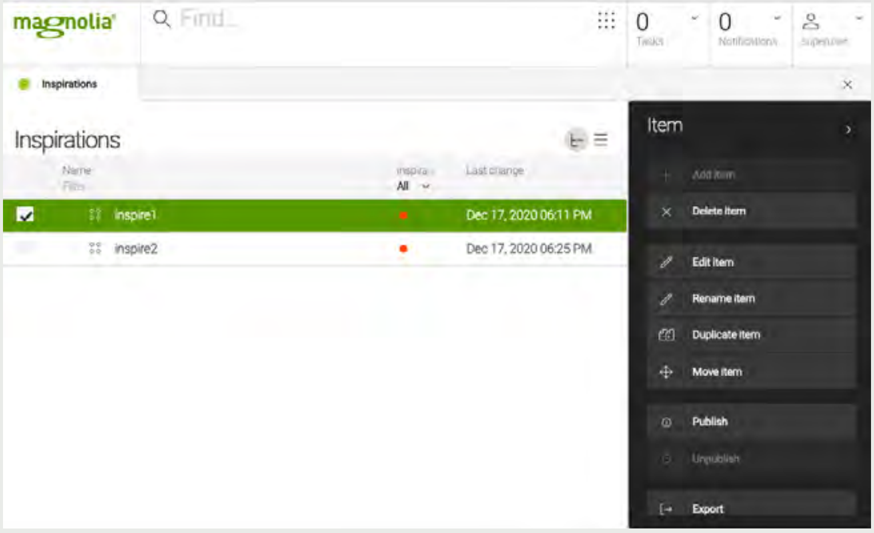


Black Flat Front Wool Suit
Black Single Pleat Athletic Fit Wool Suit
Charcoal Flat Front Athletic Fit Shadow Striped Wool Suit
Charcoal Single Pleat Striped Wool Suit
Charcoal Single Pleat Wool Suit Med
Navy Single Pleat Wool Suit

Pick and place products directly from SFCC catalogs to create content with true call-to-action capabilities

Authors can add SFCC products to the content within every Magnolia content authoring tool. Authors can showcase products in a context that highlights its benefits or tells a relevant engaging story.

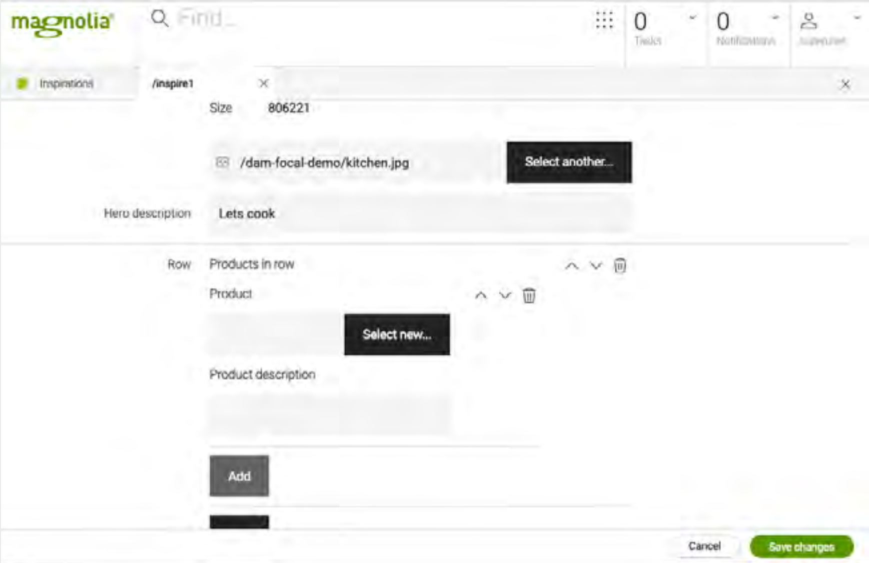
Open a Magnolia Content app. (For example, 'Inspirations' from the magnolia-sfcc demo project.)



Inspirations

Name	Inspira...	Last change
☑ inspire1	●	Dec 17, 2020 06:11 PM
inspire2	●	Dec 17, 2020 06:25 PM

Edit (or Create) an item. Choose "Select new…"



Size 806221

/dam-focal-demo/kitchen.jpg

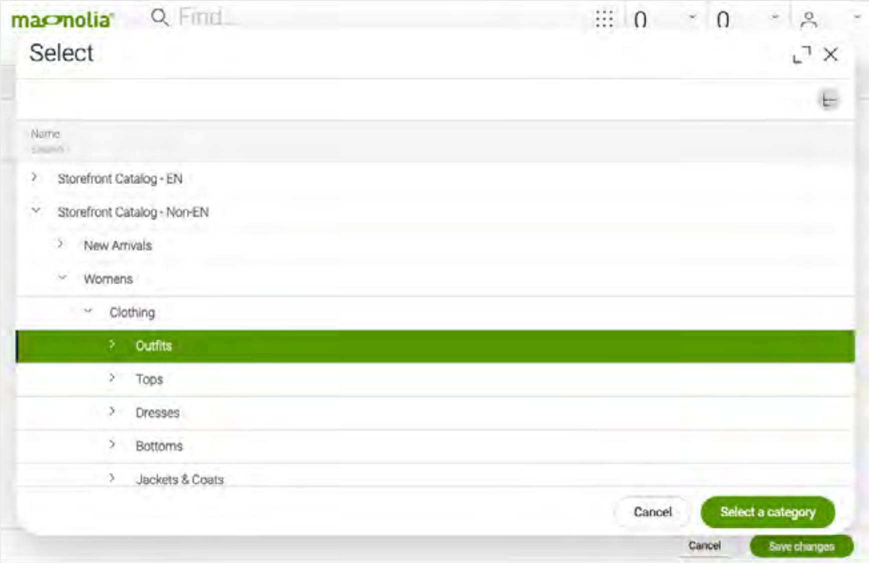
Hero description: Lets cook

Row: Products in row

Product

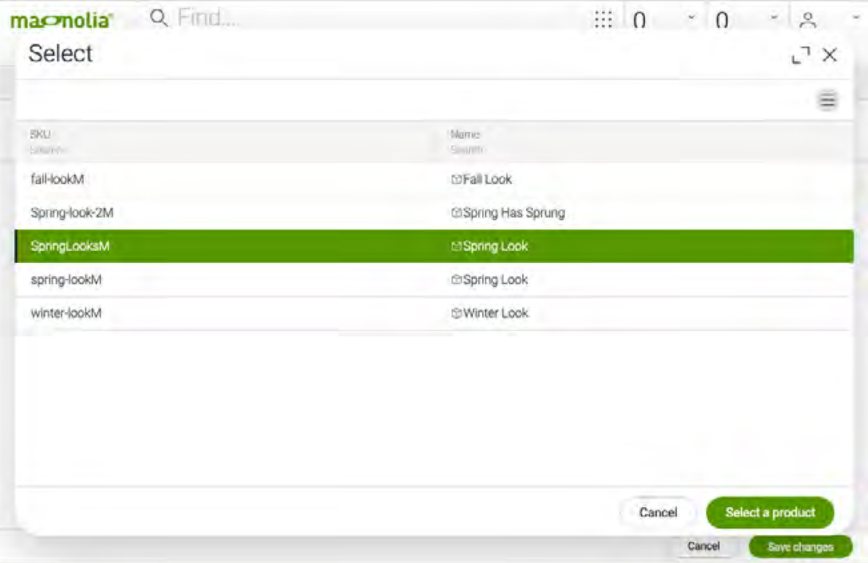
Product description

Select a catalog.



- Storefront Catalog - EN
- Storefront Catalog - Non-EN
  - New Arrivals
  - Womens
    - Clothing
      - Outfits
      - Tops
      - Dresses
      - Bottoms
      - Jackets & Coats

Select a product.

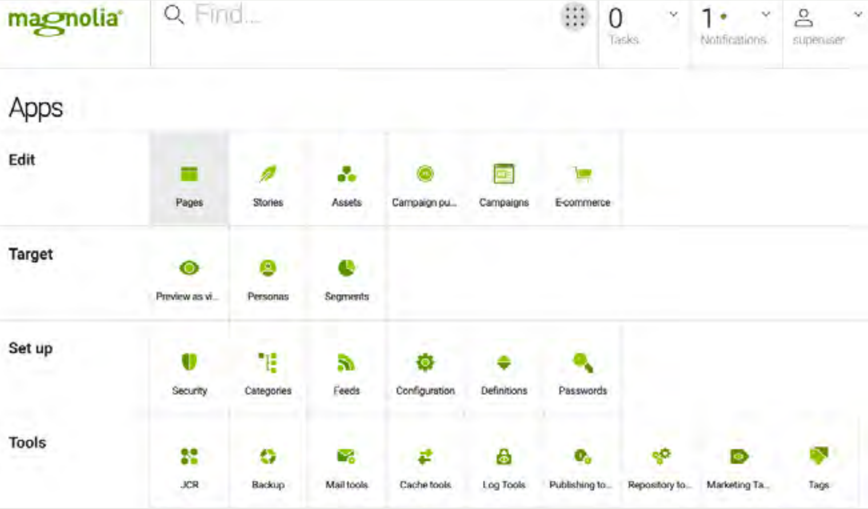


SKU	Name
fall-lookM	Fall Look
Spring-look-2M	Spring Has Sprung
SpringLooksM	Spring Look
spring-lookM	Spring Look
winter-lookM	Winter Look

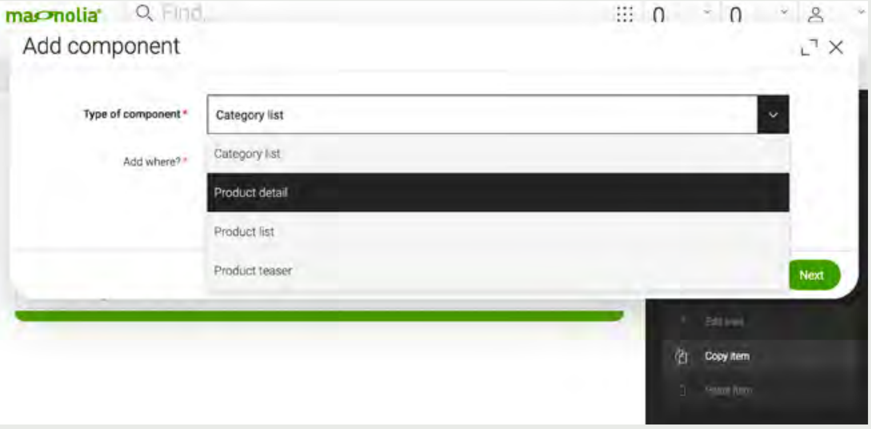
Create landing pages - Full WYSIWYG and multi device preview

Authors can pick and place SFCC products directly on pages as well. The page editor offers a high-fidelity visual preview of the final experience that will be delivered headlessly to Storefront or another frontend.
A common use case is to create the main site landing page, as well as landing pages for departments and campaigns. Authors have complete control and can bring together CMS and commerce content together on one page.

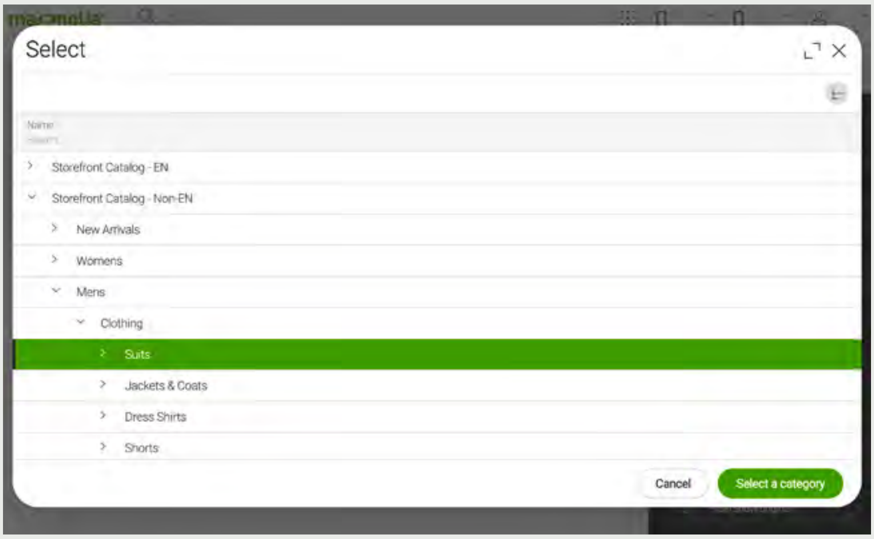
Open the Pages app.



Add a page.
On the page, add a component.
Select the 'Product Teaser' component.



Select the catalog.

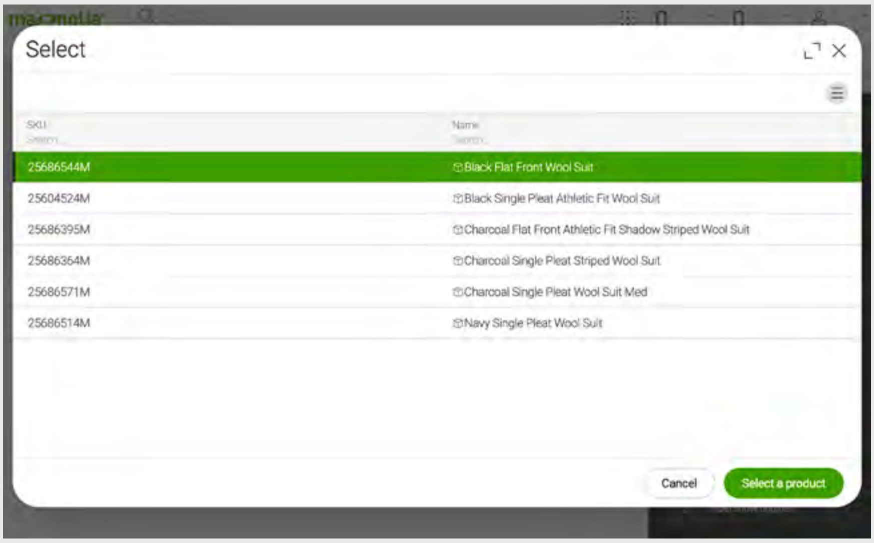


Select

Name

- Storefront Catalog - EN
- Storefront Catalog - Non-EN
  - New Arrivals
  - Womens
  - Mens
    - Clothing
      - Suits
      - Jackets & Coats
      - Dress Shirts
      - Shorts

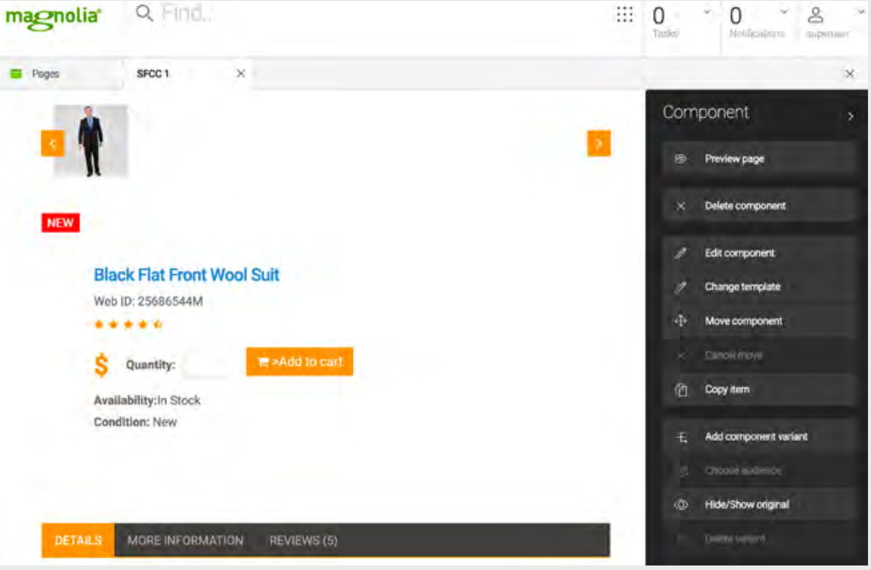
Select the product.



Select

SKU	Name
25686544M	Black Flat Front Wool Suit
25604524M	Black Single Pleat Athletic Fit Wool Suit
25686395M	Charcoal Flat Front Athletic Fit Shadow Striped Wool Suit
25686364M	Charcoal Single Pleat Striped Wool Suit
25686571M	Charcoal Single Pleat Wool Suit Med
25686514M	Navy Single Pleat Wool Suit

The product teaser is displayed on the page.



Black Flat Front Wool Suit

Web ID: 25686544M

$ **Quantity:**

Availability: In Stock

Condition: New

(Sample template. All templates are fully customizable.)

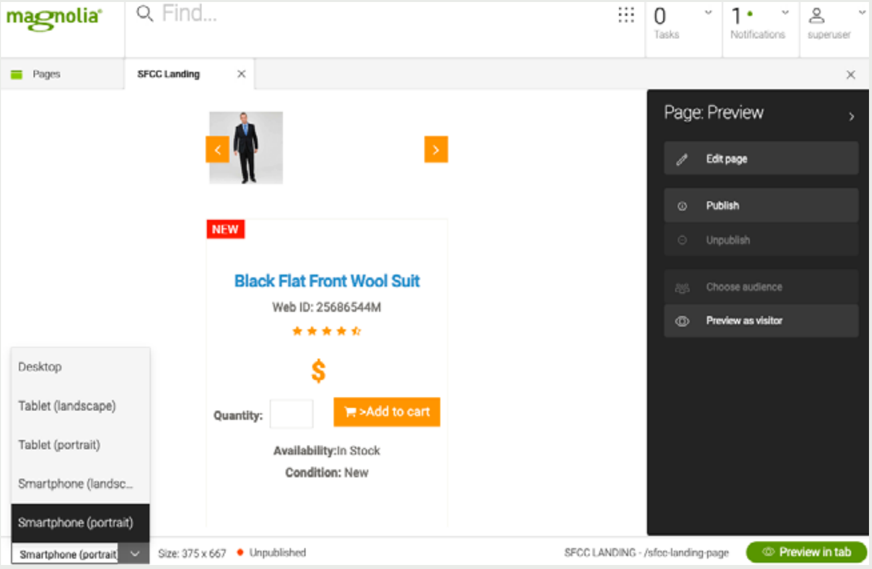
Preview before publishing

Authors need full preview capabilities to ensure their content and experience is correct. Magnolia's content APIs include a preview API so that Storefront (or other frontends) can provide a preview version of the final experience.

For content in the Pages app, authors can toggle to a preview mode which shows the content with no editing interface and which allows multi-resolution previews for mobile, tablet and desktop.

In the Pages app, click 'Preview Page'.

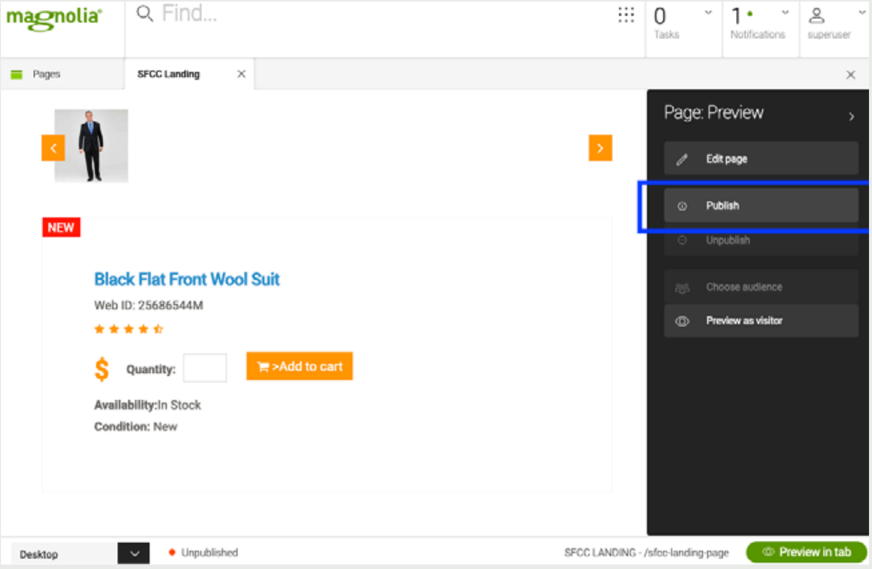
Use the device dropdown box in the lower left corner.



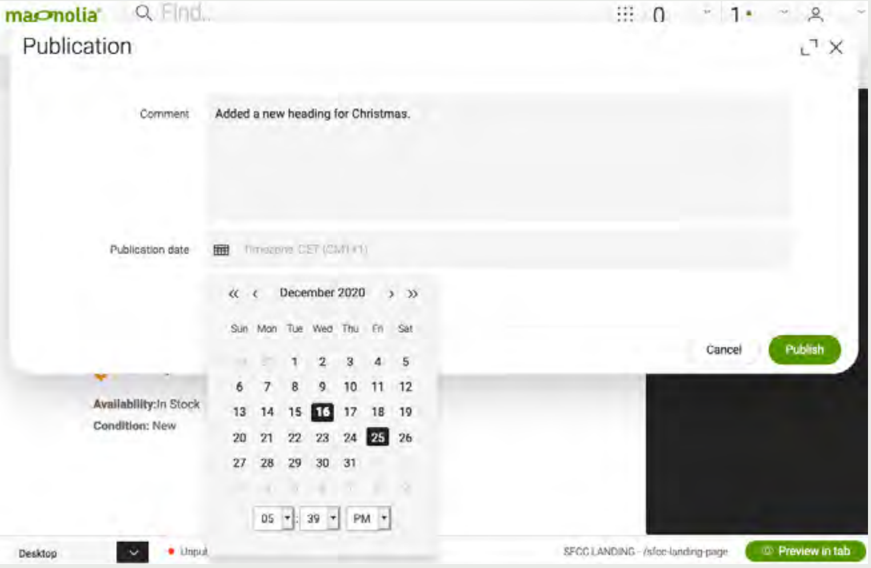
Publishing and approval workflow

Authors do their work on a so-called 'Author Instance' of Magnolia. When content is ready, they publish their changes.

In a content app or the Pages app, click the Publish action.



If an approval workflow is not active, then the content is immediately published and delivered by the content REST APIs. If an approval workflow is active, then the author can enter a comment and optionally a date for publication.



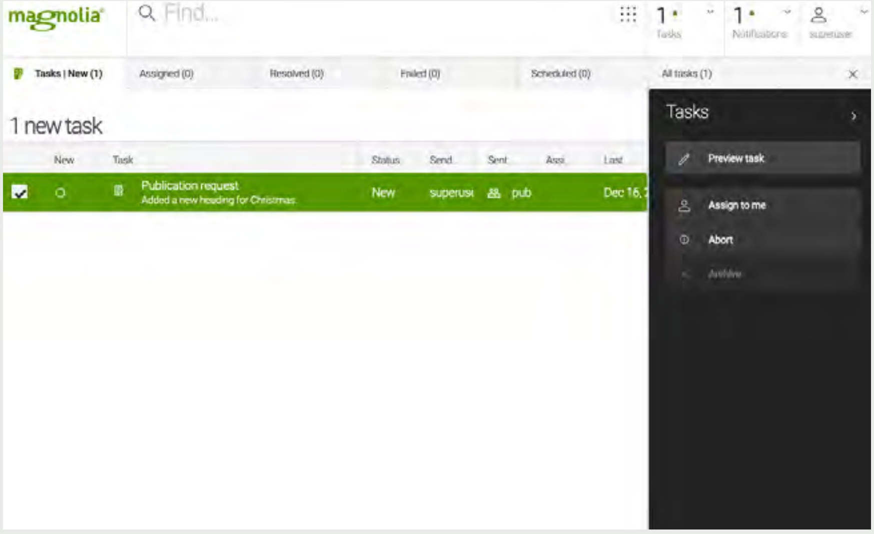
Comment: Added a new heading for Christmas.

Publication date: Timezone CET (GMT+1)

Availability: In Stock
Condition: New

Then, any user with an 'Editor' role will receive a notification that there is a publishing task waiting for their approval.

The Editor role, can review the changes and approve or reject the publication.



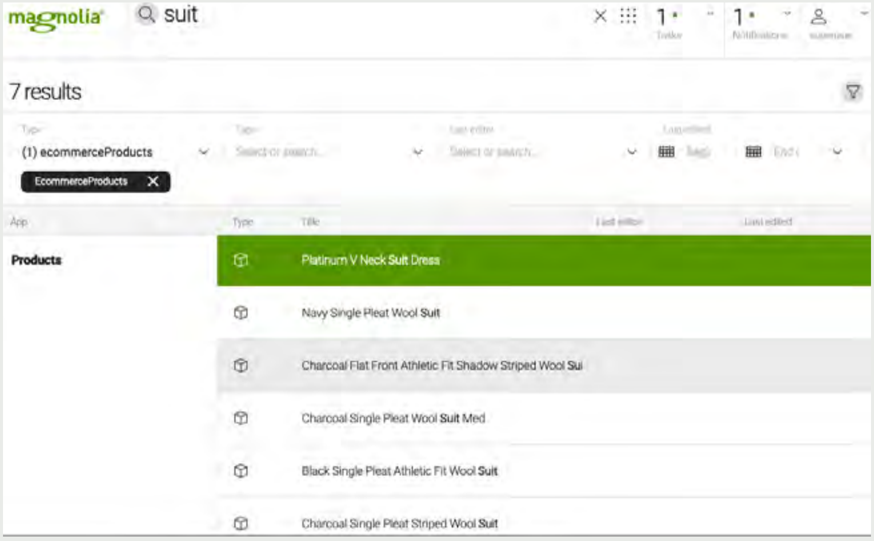
1 new task

New	Task	Status	Send	Sent	Assi	Last
○	Publication request Added a new heading for Christmas	New	superuse		pub	Dec 16, 2

Powerful Integrated search

Authors can search and find SFCC products and catalogs right in Magnolia. The search covers all content in Magnolia and all integrated systems. The search can be targeted by type, tag, editor, and date.

Enter your search term in the 'Find bar' at the top of the screen.



7 results

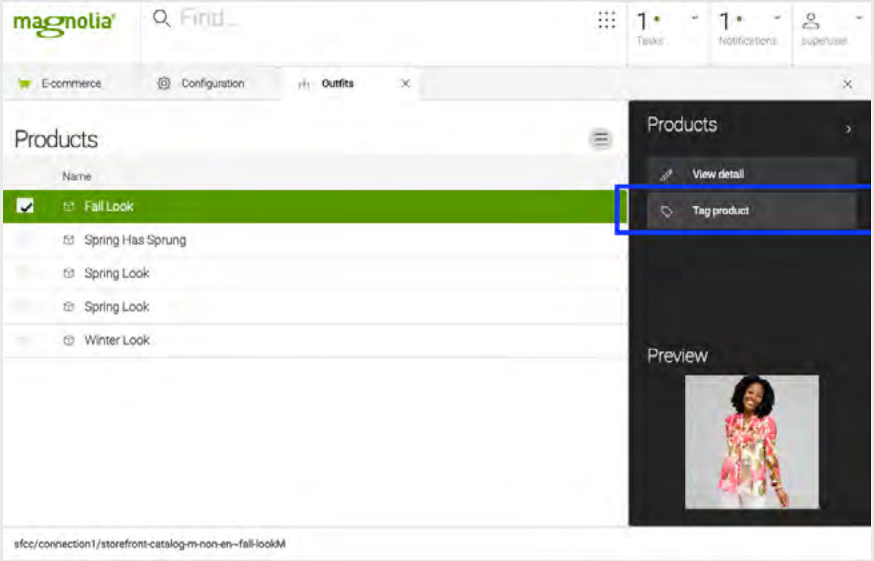
Type: (1) ecommerceProducts

App	Type	Title	Last editor	Last edited
Products		Platinum V Neck Suit Dress		
		Navy Single Pleat Wool Suit		
		Charcoal Flat Front Athletic Fit Shadow Striped Wool Sui		
		Charcoal Single Pleat Wool Suit Med		
		Black Single Pleat Athletic Fit Wool Suit		
		Charcoal Single Pleat Striped Wool Suit		

A.I. powered image recognition and text classification

Authors can trigger A.I. powered tagging in the E-commerce app with the Tag product action.
These tags make content more discoverable in the 'Find Bar', and can be used in creating dynamic lists of content in the customer experience.

Open a product in the E-commerce app.
Use the 'Tag product' action.



Advanced content management features

The integration of Magnolia and SFCC brings all of Magnolia's features to content authors.
Please see the documentation of these use cases which are highly relevant to this integration, but which are not directly involved in the integration itself.

→ Deliver relevant content, personalized to each of your visitors

→ Pick and place forms from external systems, such as Salesforce Marketing Cloud

→ Check content performance by pulling in analytics data and displaying it next to the content

→ Targeted content rollout to multiple sites based on global multilingual content

→ Create and deliver slot based campaigns and deliver fragments for product categories

→ Asset management with focal points and image hotspots. Image hotspots can directly link to SFCC product detail pages

Security Best Practices

Magnolia’s APIs are protected by classical enterprise groups / roles mechanisms. Roles carry permissions for content access as well as for API access itself.

If no authentication parameters are provided with a request then magnolia falls back to the anonymous role.In this case access is granted based on the anonymous role configuration.

Therefore the anonymous role must only grant access to public APIs/content. Everything else must be requested with authentication credentials.

You can find the documentation describing the magnolia user security mechanisms here:
→ https://documentation.magnolia-cms.com/display/DOCS/Roles+and+access+control+lists

Security best practices:
→ https://docs.magnolia-cms.com/product-docs/Administration/Security/Security-best-practices.html

Full security documentation:
→ https://docs.magnolia-cms.com/product-docs/Administration/Security.html

Release History

Version	Date	Changes
20.1.0	2021-03-20	Initial release

Get in touch

To learn how Magnolia can help you launch great digital experiences faster, contact us at:

Switzerland - Headquarters

contact@magnolia-cms.com

United States

contact-us@magnolia-cms.com

Czech Republic

contact@magnolia-cms.com

Spain

contact-es@magnolia-cms.com

United Kingdom

contact-uk@magnolia-cms.com

Vietnam

contact-apac@magnolia-cms.com

Singapore

contact-apac@magnolia-cms.com

China

contact-apac@magnolia-cms.com

www.magnolia-cms.com

© 2021 Magnolia International Ltd.