



Home



My Network



Jobs



Messaging



Notifications



Me



For Business

Get hired with
[Premium](#)

Create your own newsletter

Start your own discussion with a newsletter on LinkedIn. Share your thought leadership with every new edition.

[Try it out](#)

SAP Spartacus: Customizing the Product Details Page Part 1

By Avinash Jadhav

SAP Spartacus Customization : PDP Page

SAP Spartacus: Customizing the Product Details Page Part 1



Avinash Jadhav

SAP Composable Storefront(formerly Spartacus) Trainer / Coach / Expert |...

5 articles

✓ Following

October 20, 2022

Open Immersive Reader

Hey Guys, I hope you are doing well. Today I'm going to talk about customizing OOTB components of SAP Spartacus Storefront. Before showing you how we should approach OOTB customization, I will talk about what is happening currently in the world of **SAP Commerce Cloud** and **Spartacus**.

The companies which are using SAP Commerce Cloud or so called Hybris are now moving from **Accelerator** to **Headless architecture** with the help of Spartacus library. Because this library is based on Angular, and which is why it allows organizations to hire experienced angular folks who can easily take grasp of SAP Spartacus library.

SAP Spartacus is Angular based eCommerce storefront, which has been used by many companies till now and the one which are not moved to Headless architecture yet, they are planning to move now.

SAP Spartacus library is not well known in developers community and very few developers who knows about Spartacus library, are actually working on a project using Spartacus library. Usually what happens is companies hire angular developers for the projects based on Spartacus and

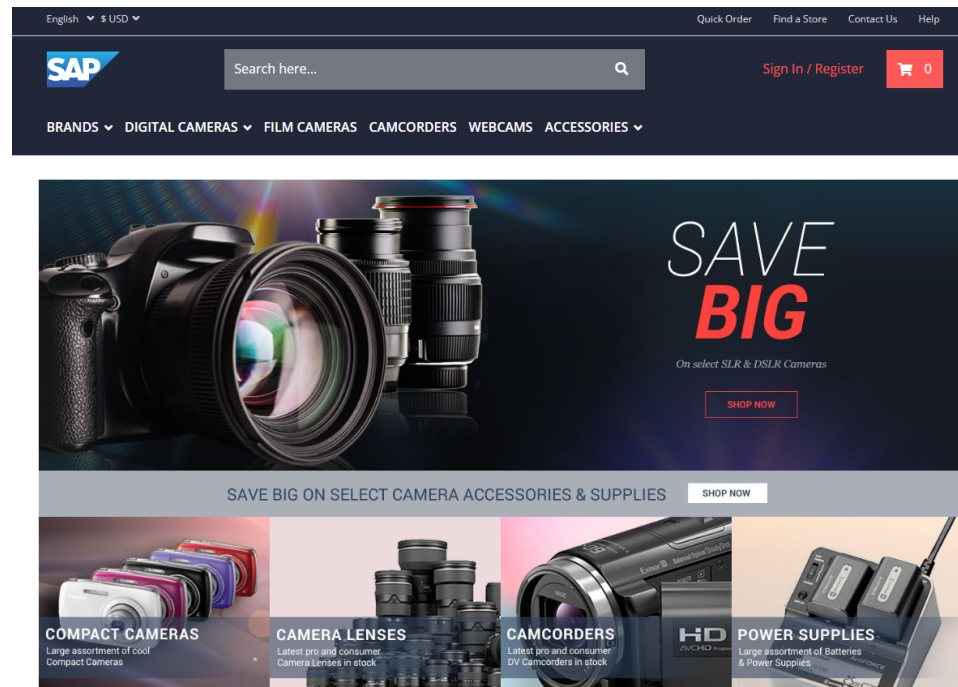
they start working on the project, and this is where many developers struggle to get the basics of Spartacus as not much of resources are available on Spartacus.

Same thing happened to me as well. Hence, I thought writing articles on basics of Spartacus library which will help developers community to make smooth transition in any Spartacus project.

Okay, enough of boilerplate content, let's focus on **OOTB** customization.

Incase, you don't know what is OOTB, it is called **out-of-the-box** which can be a feature or component which is provided by Spartacus. It is very general term used in any Spartacus project.

This is how our default storefront home page looks like:



Spartacus storefront PDP (Product details page)


English ▼ \$ USD ▼ Quick Order Find a Store Contact Us Help

SAP Search here... Sign In / Register 0

BRANDS ▼ DIGITAL CAMERAS ▼ FILM CAMERAS CAMCORDERS WEBCAMS ACCESSORIES ▼

Home / Digital Compacts / Sony

DSC-N1



★★★★★ (17) [Show reviews](#)

ID 358639

\$485.57

The Cyber-shot N1 offers 8.1 effective Megapixels with high sensitivity for reduced blur, a massive 3.0" Clear Photo LCD Plus screen with touch-screen operation and a ZEISS 3x Optical Zoom.

qty

- 1 + In stock

[Add To Cart](#)

[Sign in to add to wish list](#)

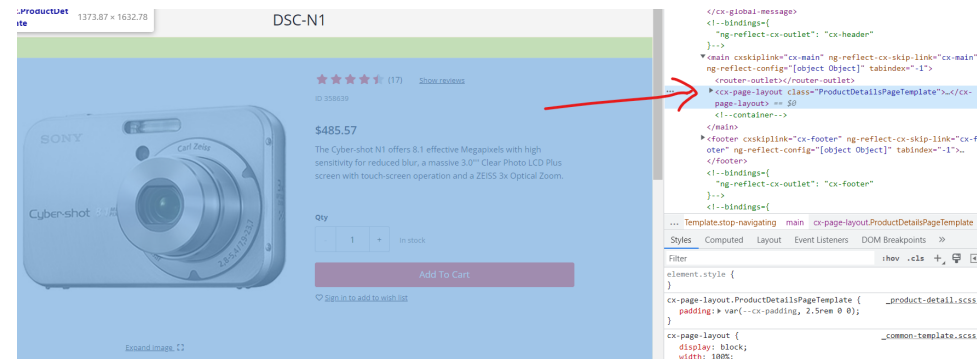
[Expand Image](#)

Note: Spartacus storefront has mapping of Pages, slots and components. Structure of every Spartacus storefront is defined in CMS in the form of pages, slots and component. This means that Spartacus UI layout mapping is tied with CMS.

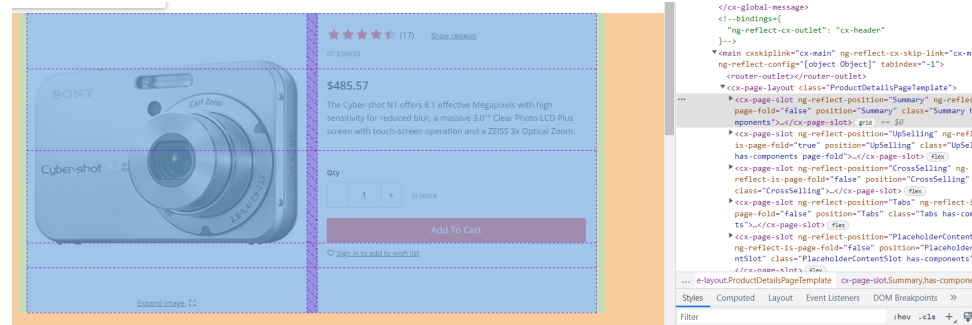
Identify CMS Component (with Element Section - Chrome Developer tool)

First step is to identify which part of the storefront we need to customize. In this case, our target is PDP page and we need to find elements and their respective CMS mapping component which is sent when we load storefront.

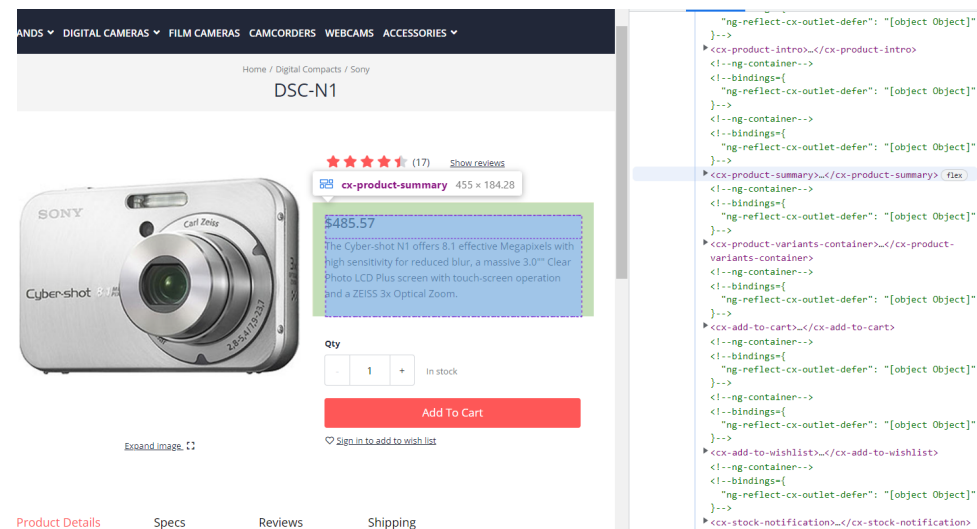
1. Find the PDP page in element section of chrome developer tool as below.



2. Inside `<cx-page-layout>` `</cx-page-layout>` element, find slot for highlighted section in below screenshot. This highlighted section is defined as **Summary slot** which is defined as `<cx-page-slot>` element. In this element position attribute contains the value "Summary" which can be used with Outlets(will cover this in separate article) as well to replace with angular component.



3. Finally identify the component to be modified.



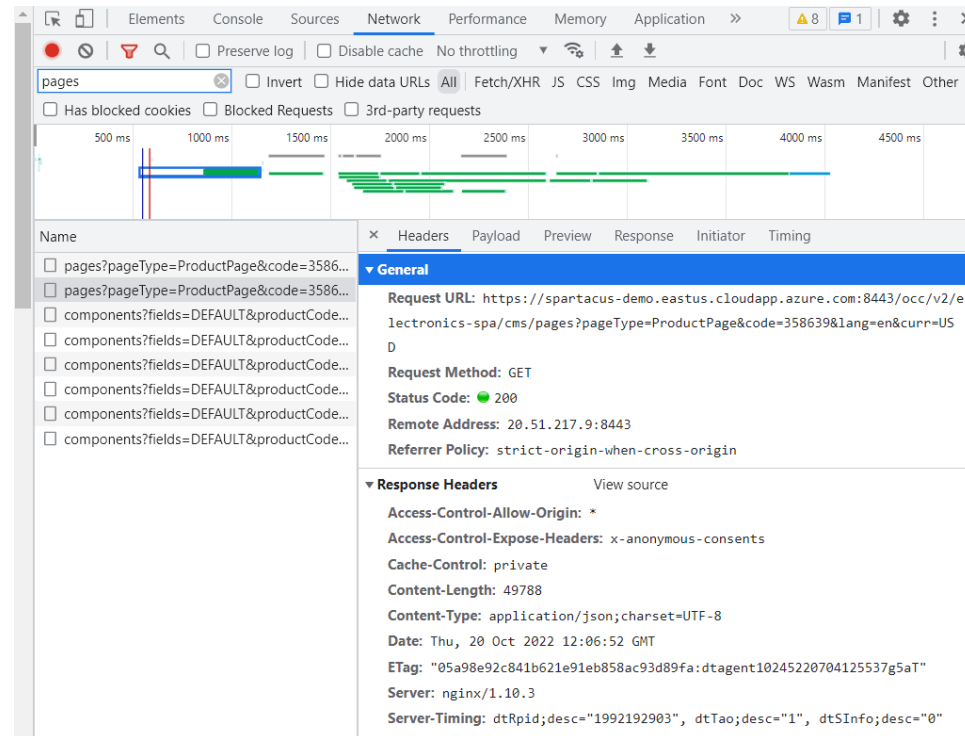
Let me explain above screenshot. As we kept nesting into cx elements from page to slot, the final place will be the components which will be replaced with actual angular components. If you see in element section there are various components like `<cx-product-intro>`, `<cx-product-`

summary>, <cx-product-variants-container>, <cx-add-to-cart>, etc.

Note: CX prefix tells us that those are created & provided by Spartacus.

Identify CMS Component (with Network tab in Chrome Developer tool)

1. Go to network tab in the developer console.
2. Load the storefront and search URL with word pages



In above screenshot, you see the API call for PDP CMS page.



Above contains Product Details page layout response.

Important properties to note are **contentSlots**, **template**, **typeCode** and **UID**. These properties can be used for mapping against custom angular components to be replaced.

Inside the property **contentSlots** you will find array of slots exist within Product Details page. If you see there are total 17 slots(array starts with zero) in it. In the 16th slot if you observe, contains properties like components, position, slotId, etc. Important one to use are **components** and **position**

```

▼ {uid: "productDetails",...}
  ▼ contentSlots: {...}
    ▼ contentSlot: [{slotId: "HeaderLinksSlot",...}, {slotId: "TopHeaderSlot",...}, {slotId: "BottomHeaderSlot",...}]
      ▶ 0: {slotId: "HeaderLinksSlot",...}
      ▶ 1: {slotId: "TopHeaderSlot",...}
      ▶ 2: {slotId: "BottomHeaderSlot",...}
      ▶ 3: {slotId: "CrossSellingSlot",...}
      ▶ 4: {slotId: "SiteContextSlot",...}
      ▶ 5: {slotId: "SiteLogoSlot",...}
      ▶ 6: {slotId: "UpSellingSlot",...}
      ▶ 7: {slotId: "SiteLinksSlot",...}
      ▶ 8: {slotId: "PlaceholderContentSlot",...}
      ▶ 9: {slotId: "SearchBoxSlot",...}
      ▶ 10: {slotId: "NavigationBarSlot",...}
      ▶ 11: {slotId: "MiniCartSlot",...}
      ▶ 12: {slotId: "Section3-ProductDetailsSlot",...}
      ▶ 13: {slotId: "FooterSlot",...}
      ▶ 14: {slotId: "TabsSlot",...}
      ▼ 15: {slotId: "ProductSummarySlot",...}
        ▶ components: {component: [{uid: "ProductImagesComponent",...}, {uid: "ProductIntroComponent",...},...]}
          name: "Summary for product details"
          position: "Summary"
          slotId: "ProductSummarySlot"
          slotShared: true
          slotUUid: "eyJpdGVtSWQ1Oi1JQcm9kdWN0U3VtbWVyeVNsbnB3Q1LCJjYXRhbG9nSWQ1Oi1JbGVJdHJvbmJjcy1zcGFDb250ZWZl"
        ▶ 16: {slotId: "HomepageNavLinkSlot",...}

```

The slot that we are going to target is **ProductSummarySlot**.

```

▼ 15: {slotId: "ProductSummarySlot",...}
  ▼ components: [{component: [{uid: "ProductImagesComponent",...}, {uid: "ProductIntroComponent",...},...]}
    ▼ component: [{uid: "ProductImagesComponent",...}, {uid: "ProductIntroComponent",...},...]}
      ► 0: {uid: "ProductImagesComponent",...}
      ▼ 1: {uid: "ProductIntroComponent",...}
        container: "false"
        flexType: "ProductIntroComponent"
        modifiedtime: "2022-08-21T20:58:28.621Z"
        name: "ProductIntroComponent"
        synchronizationBlocked: "false"
        typeCode: "CMSFlexComponent"
        uid: "ProductIntroComponent"
        uuid: "eyJpdGVtSWQlOiJQcm9kdWN0SW50cm9Db21wb251bnQlICJYXRhbG9nSWQlOiJlbGVjdHJvbm1jcy1zcGFDb2"
      ► 2: {uid: "QualtricsEmbeddedFeedbackComponent",...}
      ► 3: {uid: "ProductSummaryComponent",...}
      ► 4: {uid: "VariantSelector",...}
      ► 5: {uid: "AddToCart",...}
      ► 6: {uid: "ConfigureProductComponent",...}
      ► 7: {uid: "AddToWishListComponent",...}
      ► 8: {uid: "StockNotificationComponent",...}

```

And finally we are now entering into component section where we are going to integrate our custom component. If

you see the **ProductIntroComponent** important properties to note are **flexType** and **typeCode**. We are going to use **flexType** inside our angular CMS mapping to replace with our custom angular component.

And that's it for this article! It would be lengthy article if I try to cover coding implementation as well, hence we will cover that in second part of this **SAP Spartacus PDP customization article series**.

I hope you have enjoyed my article and learned something. Please share this article with your friends, colleagues, team mates and help them too.

Sharing will motivate me to write more and more on this topic! Cheers!

[Report this](#)

Published by



Avinash Jadhav

SAP Composable Storefront(formerly Spartacus) Trainer / Coach / Expert | Ang...
Published • 1y

5
articles

✓ Following

NEW ARTICLE is live on SAP Spartacus Customization! [#sap](#) [#spartacus](#)
[#sapcommercecloud](#) [#sapcommerce](#) [#storefront](#) [#sapcommunity](#) [#headless](#)
[#headlesscommerce](#)



Like



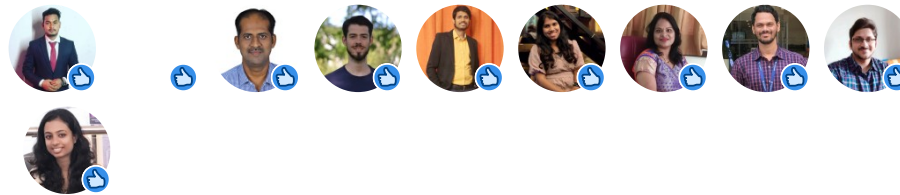
Comment



Share

10 3 comments

Reactions



3 Comments



Add a comment...



Samata Adsul Jadhav (She/Her) • 2nd

UI/UX Designer

1y ...

Great article 👍

Like | Reply



Vallabh Bothre • 2nd

IT Professional | Project Manager | Problem Solving Skills | Leadership |
Mentor | Empowering people | Educator | Author | Engineer

1y ...

Wonderful avinash. The article is really amazing. Waiting for more!!!

Like | Reply

Load more comments



Avinash Jadhav

SAP Composable Storefront(formerly Spartacus) Trainer / Coach / Expert | Angular | React.js

✓ Following

More from Avinash Jadhav

SAP Spartacus

Customizing Your Store with LayoutConfig

Customizing Your Store with
LayoutConfig in SAP
Spartacus

Avinash Jadhav on LinkedIn

SAP Spartacus

The Key to Greater Success as an Angular Developer

SAP Spartacus: The Key to
Greater Success as an
Angular Developer

Avinash Jadhav on LinkedIn

SAP Spartacus: Customizing the Product Details Page Part 2

By Avinash Jadhav

SAP Spartacus: Customizing
the Product Details Page Part
2

Avinash Jadhav on LinkedIn

[See all 5 articles](#)

About

Community Guidelines

Privacy & Terms ▼

Sales Solutions

Safety Center

Accessibility

Careers

Ad Choices


Mobile


Talent Solutions


Marketing Solutions

Advertising

Small Business

 **Questions?**
Visit our Help Center.

 **Manage your account and privacy**
Go to your Settings.

 **Recommendation transparency**
Learn more about Recommended Content.

Select Language

English (English)