

 Launch Package Standalone



Session Objectives

- △ Review the basics of the Catalyst framework and set it up via One-Click Catalyst
- △ Introduce the Makeswift Visual Builder as an alternative to the native BigCommerce design experience

Introductions/Discovery Questions

How are you doing? How are you feeling about the store and the process so far?

Have you had the chance to review any of the topics we covered last time?

In this session we will be walking through the design of your store using the Catalyst framework with the Makeswift Visual Builder.



Coaching Topics

Introduction to Catalyst - (unknown) min

Reference Material:

- [Catalyst for Composable Commerce](#) (BigCommerce KB)

Prerequisites:

- Multi-Storefront enabled BigCommerce Control panel with at least one available seat.
- One-Click Catalyst available in the Channels menu.

Introduction to Catalyst

- *Catalyst is a next generation approach to Composable Commerce, providing a fully customizable UI kit and a comprehensive GraphQL Storefront API client, optimized for Next.js and React Server Components. It can be used standalone or with the Makeswift Visual Builder.*
 - *For non-technical users, it is often sufficient to say that Catalyst is primarily the underlying code and resources that support the design from the back-end to provide a stunning front-end experience, which can all be customized by a developer to meet the needs of the project.*
 - *The Makeswift Visual Builder puts further power into the hands of non-technical users to be able to customize the design on top of the work that developers have done.*

Who should use Catalyst

- *Ideally, Catalyst is a good fit for merchants who*
 - *Have developers on staff to configure Catalyst beyond the basic setup and connect their storefront with the Catalyst framework.*
 - **Note:** As Catalyst is built with Next.js, those developers should have some experience with Next.js for best results
 - *Explicitly intend to have a headless implementation on BigCommerce*
 - *Are prepared to host their Catalyst implementation off of BigCommerce*
- *B2B merchants looking to use Catalyst can do so easily with Buyer Portal as it renders*



directly on top of Catalyst storefronts.

△ One-Click Catalyst

- *In order to facilitate rapid deployment and decrease the barrier to entry, BigCommerce provides a quick One-Click setup for Catalyst in the control panel that allows access to the most recent build of the Catalyst framework as well as the Makeswift Visual Builder.*
 - **NOTE:** Even using the One-Click Catalyst setup option, merchants will currently be required to host their Catalyst implementation on a third-party system. Currently BigCommerce configures that automatically using Vercel, but as the One-Click Catalyst deploys to a BigCommerce sandbox account merchants should migrate to their own hosting (either their own Vercel account, or another system if desired) for their live site.
- In the control panel navigate to Channels
 - Click **Create Channel** or **Add new**
 - In the Storefronts section, locate Catalyst and click **Create**. This will open the Catalyst app and prompt the user for settings.
 - Provide a storefront name
 - Select whether to use sample data or an empty channel
 - Sample data will add new product data and several basic pages to the store in order to support the new channel. This will provide a reference point for various features of the system.
 - Empty channels will not create any new product data and will simply use the data that is already in place.
 - **Note:** Using the sample data is the recommended approach, as some of the default Makeswift template data is tied to the sample product data.
 - Select a primary language
 - Click Create
- *The next step can take a few minutes as the current build of Catalyst is being configured for use on a new channel that is created in the process. Once the channel is configured, it can be accessed either from Channels or Apps.*
- **NOTE:** Current behavior for One-Click Catalyst does not force the store into Multi-Storefront. As such, the current recommended sequence for Single-Storefront stores is
 - Navigate to Channels > Create Channel > Enable Storefront Seats.
 - Opt into all MSF functionality.



- Follow steps for One-Click Catalyst.
- **NOTE:** When new builds of Catalyst are available, the storefront can be refreshed using the Redeploy option from the Actions dropdown in the channel overview.
- *The Actions dropdown will also allow the user to view deployment logs, which include the current versions of various components, warnings, and the specific information for the implementation.*
- *A Catalyst channel may be deleted from the overview page in the Actions dropdown.*

Makeswift - (unknown) min

Reference Material:

- [Makeswift Documentation](#)
- [Components - Makeswift](#)
- [Catalyst Components in Makeswift](#)

▴ Makeswift High-Level Overview

- *Makeswift is first and foremost a website builder, designed to compete with any website builder out of the box, and is deeply integrated with BigCommerce to support the use of Catalyst for composable commerce.*
- *Makeswift is fully composable and provides both a minimal code requirement and the freedom to employ code as necessary to enhance front-end UX/UI.*

▴ Low-Code-First Flow

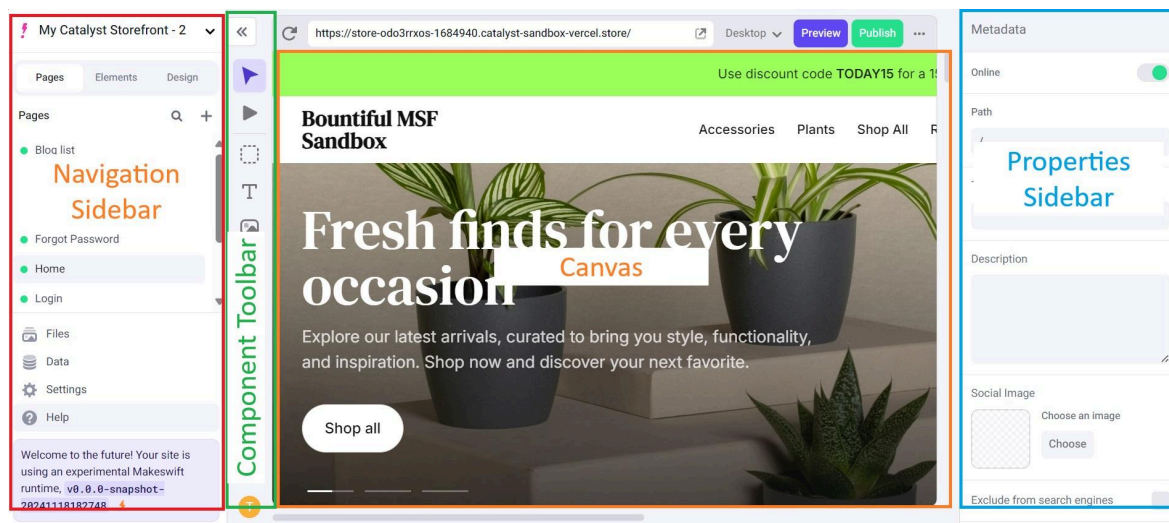
- *To support the Low-Code-First experience, allowing new users to familiarise themselves with Makeswift and support those teams with less technical experience, BigCommerce provides a One-Click setup for Catalyst that incorporates Makeswift by default.*
- Access Makeswift via Channel Manager
 - Once the [One-Click Catalyst channel creation](#) workflow is completed, open the channel overview either through the One-Click Catalyst app or by navigating to Channels and clicking on the storefront you wish to edit.
 - Click **Edit in Makeswift** to access the Makeswift Visual Builder
 - If this is a user's first time accessing Makeswift, they will be required to connect to their Makeswift workspace with their BigCommerce login to access the site builder. This only needs to be completed once, regardless of how many Catalyst



storefronts the user has access to.

- **Note:** A Makeswift account should be automatically created on the back-end. The connection step ensures the user is able to seamlessly access the workspace from the control panel going forward.
- **Note:** All remaining info is for the Low-Code structure specifically.

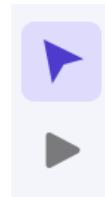
Makeswift Editor Basics



- The editor is broken down into five main regions.
 - The **Navigation Sidebar** allows access to the various pages, elements, and design features.
 - The **Component Toolbar** is where components live. Some components will be available out of the box from both Catalyst and Makeswift, but any React component can be turned into a component.
 - **NOTE:** Components are similar to widgets in the native BigCommerce Page Builder, but they have more customisability generally and may be configured in Catalyst via code or in Makeswift.
 - The **Canvas** is the section where you can drag-and-drop components, edit features in place, and see how the design of the page changes based on the choices you make.
 - The **Properties Sidebar** will display content based on what is selected both in the Navigation Sidebar and in the Canvas itself. This area allows you to adjust the SEO, visual properties, shared component features, and other general properties.



- *If a theme is used, some of the design elements will be selectable directly in this area.*
 - *The last area is the Control Bar at the top, which includes the current page and form-factor breakpoint as well as the buttons for previewing and publishing changes.*
 - **NOTE:** The default Makeswift configuration using One-Click Catalyst sets up a free account, which is limited to 5 Publish events per account per month. As such, Publishes should be used sparingly while using the default configuration, but publishing only once per week will generally keep users on the free plan under that limit.
 - **NOTE:** Changes in Makeswift are automatically saved, but they are not pushed to the live site until a publish event.
 - **Note:** The Preview button allows users to share preview links with guest users on your account. Adding this to the workflow may help to manage publishes by allowing for approval prior to roll-out. See [Guest Users](#) and [Previewing](#) for more information.
- *In the Component Toolbar, there are two arrow buttons toward the top, one resembling a cursor and the other resembling a play button.*
 - *The cursor style button (top) is the Build selector, which switches the page into Build Mode, allowing for components to be added, removed, or otherwise adjusted.*
 - *The play button (bottom) is the Interact selector, which switches the page into Interact Mode, allowing the user to test user experience, navigation, and other features.*
 - **NOTE:** Currently, while in Interact Mode, the active page does not update, so changes made to other pages will not carry through to those pages. Best practice is to explicitly change the active page using either the Control Bar at the top or the Navigation Sidebar at the left.

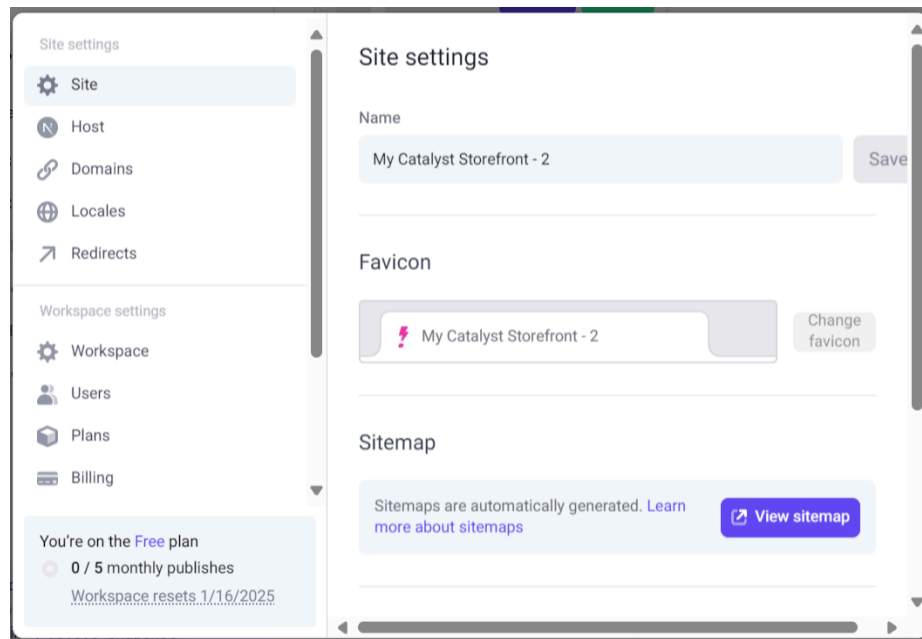


Settings

- *The settings menu is available to adjust both Site and Workspace core behavior. This includes Home Page Title, Favicon, Sitemap, and several other features that are common necessities in web design generally and eCommerce particularly.*
 - **Site Settings**
 - *Site*



- Includes Page Title, Favicon, Archive, and Delete options.
- Archive will take a site down so it is not available.
- Delete will remove the site's data from the server.



- **Host**
 - Includes the Host URL (where the site is currently being hosted) and the Site API Key (which gives access to API functionality)
- **Domains**
 - List of domains at which the site should be delivered.
 - Domains listed must be configured at their DNS host(s).
- **Locales**
 - Allows configuration of language locales per domain if desired.
- **Redirects**
 - List of redirects for pages that have been moved or deleted.
 - Redirects may be configured statically to a path or dynamically to a specific page.
 - Redirects may be labelled as temporary, if desired.
 - Note: Redirects may be auto-generated for pages when the URL is updated. Makeswift will display a dialog to opt into this behavior.

- **Workspace Settings**



- *Workspace*
 - *Change the name of the workspace.*
- *Users*
 - *Manage the users with access to the workspace.*
 - *Can add new users with specific permissions.*
 - *Number of users is limited by plan.*
- *Plans*
 - *Manage your current Workspace Plan, allowing you to add more features and reduce account restrictions.*
- *Billing*
 - *Get billing information including current tier, payment methods, tax information, and invoice history.*
- *Usage*
 - *Get information about plan limits and how much you have used against those limits.*
- *Integrations*
 - *List and manage available integrations.*
- **User Settings**
 - *Profile*
 - *Update Name, Company, and personal data with respect to your specific Makeswift login.*
 - *If a user chooses to update their information, both First and Last Name must be filled in.*
 - **NOTE:** If a user creates their Makeswift account using an automated tool (e.g. Login with Google), they will not be able to change their email address or password.

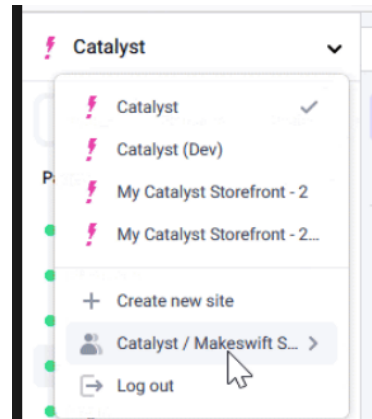
△ Sites and Workspaces

- *Makeswift's editor breaks data down into two main parts: Workspaces and Sites. These may or may not have connected features and functionality.*
 - *Workspace - The entirety of the data available to your account when a user logs in to Makeswift, which contains all files, sites, components, etc, available to them.*
 - *Workspace information is available from any Site within the Workspace.*
 - *Site - Data restricted to a specific design space, domain, or storefront*



(depending on many factors) including pages, files, data, and design choices.

- To change Sites, use the dropdown at the top of the Navigation Sidebar.
- New Sites may also be created from the dropdown menu
- Information created on a site level is generally only available on that specific site. The exception is a direct link to the data cross-site, which can pose security concerns if not managed properly for some elements such as scripts.
- *Within the Workspace or within a specific Site are several data types.*
 - Pages - Specific content pages, from informational pages and blog entries to product and category pages.
 - Not all pages of a site will show up in the Navigation Sidebar, but they still may be managed in Makeswift by putting the URL stem in the navigation bar.
 - Files - Images, icons, and other uploadable content that is to be made available throughout a Site or a Workspace.
 - Currently Makeswift does not have a broad restriction on what can be uploaded, but it does restrict many code files to prevent security issues such as Cross-Site Scripting (XSS)
 - Data - Tables whose entries may be used throughout a Site or Workspace
 - Components - Drag-and-droppable content blocks built either in Catalyst or Makeswift.
 - Components are generally Site-specific.
- *The Site Selection dropdown also has a Workspace selector to allow a user to switch between any Workspace their account is associated with.*



Components

- *Makeswift makes use of components, which are drag-and-droppable and can be used and customized for every level of design from page, to position, to form-factor.*
- *Components are informally broken down into three groups within Makeswift*
 - *Built-in Components - React components with predefined controls, determined by the developer in code. They can be modified through*





- code to add or change functionality.
- *Visually Editable (Global) Components* - Components built within Makeswift and saved to a site for later or continued use. These are an extension of the REACT Components and serve as plug-and-play reusable design units.
 - Changes to these are typically global but can be detached from the global prototype on any given page.
 - *Built-in Elements* - Elements pre-built into a page by the developer such as the global header, footer, and theme. These Components may or may not be editable within Makeswift, but they are not available to add or remove within the Visual Builder.
 - The basic components available in Makeswift are Box, Text, Image, and Button, but there are many more advanced components available within the components selection menu (the + in the Component Toolbar).
 - *Box* - Similar to the Layout widget in the native BigCommerce widget suite. This provides a work area for content or other components to live inside. It is not explicitly necessary for design, but can be used to help maintain a level of organisation.
 - *Text* - Provides an area in which rich text (including formatting) may be added to the page.
 - *Image* - Provides a block in which an image may be added. All standard HTML image properties may be edited in the Properties Sidebar.
 - *Button* - Provides a button for interactivity. Basic button behavior may be controlled in the Properties Sidebar, but advanced behavior may be added using code.
 - More advanced information on components can be found in the Makeswift docs here: [Components - Makeswift](#)
 - Once a component has been dropped into the Canvas area, it may be moved around with standard click-and-drag functionality.
 - Components that are expected to have a high reuse rate (or any reuse) may be converted into global components for rapid redeployment.
 - In the Properties Sidebar, click the cube button.
 - Provide an easily identifiable name.
 - Note: Components may be named with / characters to provide a





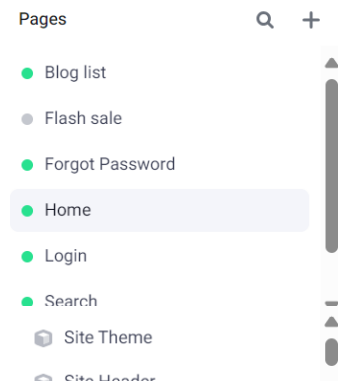
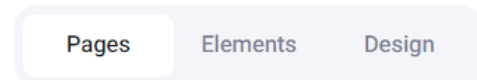


structure to organise them. For example **My/Global/Button** would create the global component in the group **My** under the subgroup **Global** with the name **Button**

- Click **Yes, continue**
- The component will now be accessible from the Component Tray (opened with the + button in the components menu).
- *Global Components carry their properties to any page in which they are used, although they may be customised on a page-by-page basis if they are Detached on any given page.*
- **Note:** Since all components and pages have separate features in the Properties Sidebar, including details for everything would take a lot of extra time. It is advised to take a moment to demonstrate one or two components while demonstrating drag-and-drop as well as click-and-drag functionality.

△ Pages, Elements, Design - Navigation Sidebar

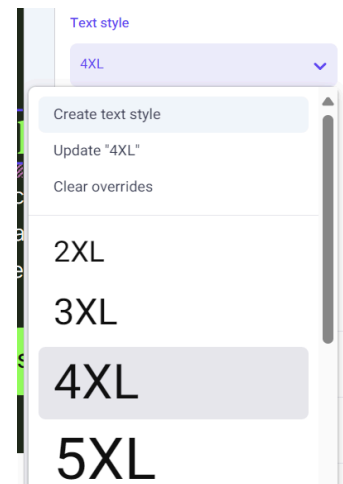
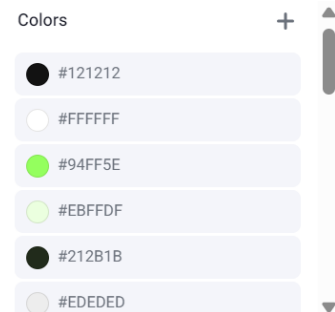
- *The Navigation Sidebar has three tabs available to manage various aspects of site design as efficiently as possible*
 - *Pages - Content pages created within Makeswift. Largely informational pages with static data that can be edited at will, but does not dynamically load.*
 - *Clicking on a page in the list will load it in the Canvas and set it as the active page for editing, allowing seamless movement to and editing of various parts of the site.*
 - *The + button in the Pages list lets you create new pages whose data will be managed in the Properties sidebar.*
 - *Pages with a grey dot  next to their name are currently unavailable outside the editor.*
 - *Pages with a green dot  are live and visible on the site's front-end.*
 - **Note:** Dynamic Components and Pages are possible with some code, and several are included in the sample data already.
 - *Elements - Hierarchical page structure including fixed and editable components, inheritance, and top-to-bottom layout (assuming no design code)*





affects the natural order).

- Clicking an element in the Elements List will do three things:
 - Select the element in-page for editing
 - Open the element's context in the Properties Sidebar
 - Scroll to the element on the Canvas
 - Adding an element may be done either by adding a new Component to the page via drag-and-drop, or by using code to adjust the default layout.
 - Built-in elements may not be deleted from the Elements list. In order to delete layout Elements, code is required.
 - NOTE: The Elements tab is very similar in form and function to the Layers view in the native BigCommerce Page Builder
- Design - General theme choices such as colors and font selections that can be referenced from around the Site.
- Changes made in this area affect any Element or Component using the asset changed.
 - Assets defined in this area will be the default selections in the Properties Sidebar.
 - **NOTE:** Deleting any design asset from the Design tab will unlink it from all elements that use the asset, which may result in inconsistent design.
 - New design assets may be added either in the Design tab or in the Properties Sidebar, at which point they will become linkable in the latter area.
 - Duplicate design assets may be created, but at present there is no way to label some design assets that are the same but have different uses that may not align in the future.
 - Text styles may be designed for size, color, and typeface, as well as several other options.
 - Text styles are currently only usable with Components and not more general Elements.
 - Text styles are more easily updated,






created, and overridden per Component.

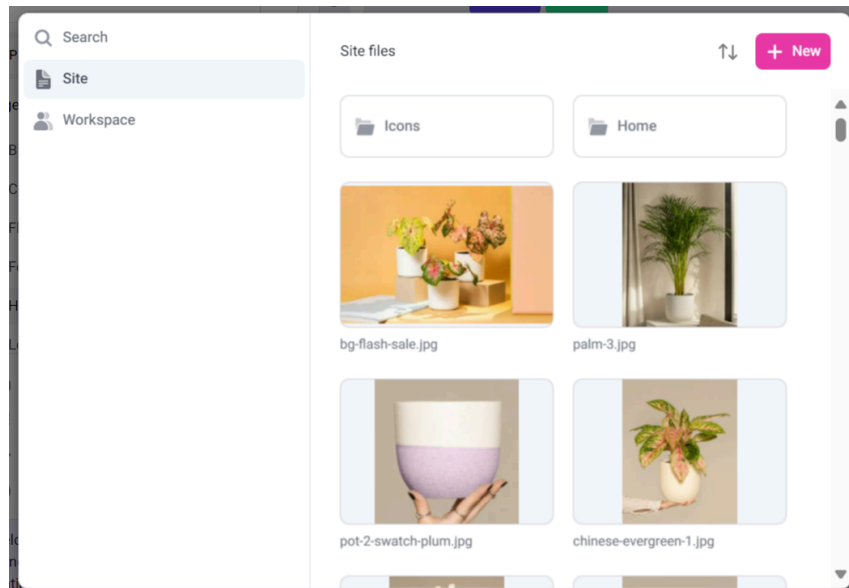
Files and Data - Navigation Sidebar

- In addition to Pages, Elements, and Design, the Navigation sidebar allows users to actively manage the Files and Data for both their Site and their Workspace.

 Files


 Data

- Files can be sorted into folders, subfolders, etc.
 - These are managed like any standard File Explorer with an experience similar to Google Drive



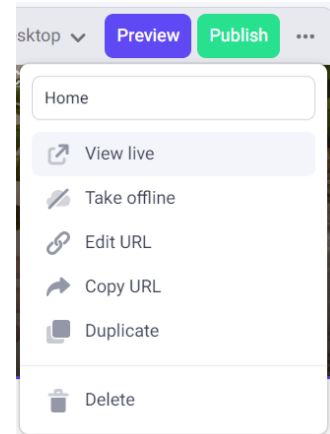
- There are no filetype restrictions inherent to Makeswift, though some may be blocked on the server end from the hosting platform.
 - Example: BigCommerce blocks the uploading of PHP files via WebDAV due to security risks.
- While file URLs may be copied directly from the File menu, these may not be the best to use for reference on the storefront. Instead, files should be selected from the relevant menus in the Properties Sidebar.

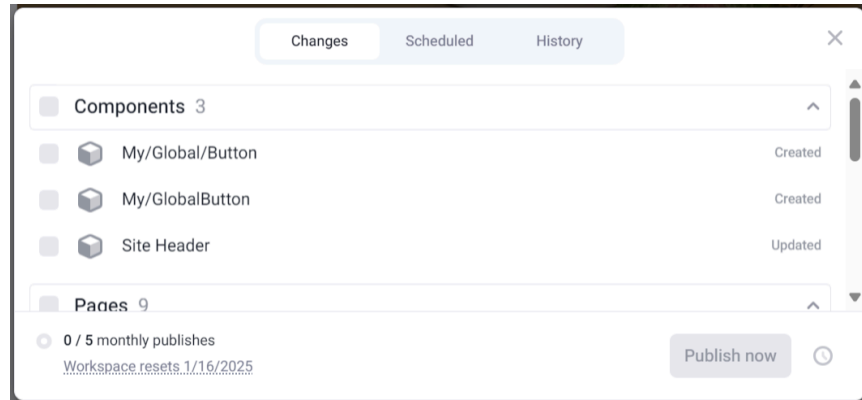
Publishing and Page Actions - Control Bar

- While editing a page or managing the site design as a whole, the Control Bar offers several options via the  button, most of which are self-explanatory.



- **View Live** shows an active preview of the page as it is currently available to customers.
- **Take Offline** makes it so the page is no longer available on the front-end of the site. It may still show up in static menus.
- **Edit URL** lets the user redefine the URL stem with which the customer would access the page, as well as the URL used within Makeswift for design.
- **Copy URL** copies the URL of the page as it would be used by customers to access the page.
- **Duplicate** makes a copy of the page.
- **Delete** removes the page from the site. This action cannot be undone.
- The URL box in the Control Bar may be used to navigate to any page on the site for editing.
 - The domain cannot be changed, only the stem.
 - If the user is in Interact Mode, changing the page via this box will return the user to Build Mode and switch the active page.
- The dropdown next to the **Preview** button allows the user to switch to other form-factors to manage design for other screen types.
 - Changes made to an upstream form-factor pass downstream to form-factors below them (Desktop -> XL -> Large -> Medium -> Small)
 - Changes made to a downstream form-factor do not pass back upstream. (Desktop -> Large, but not the other way)
 - The content of an element is not part of the downstream flow, only the design factors (size, position, visibility, etc)
- The **Publish** button opens the dialog for publishing changes to make them live for the customer.





- *The Changes tab shows all changes that have been made to the current Site.*
 - *Changes must be selected to include them in the current Publish action.*
 - *Any changes not selected will still be available for future Publish actions.*
 - *Changes are split between the several data types, including new pages, updated pages, new global components, and updates to components and layouts.*
 - *Clicking the checkbox for a given data type will select all Changes in that group.*
 - *Groups selected this way must be selected one group at a time.*
- *On higher tier plans with Makeswift, Publish events can be scheduled, and history may be viewed.*
- *It is imperative to note the “monthly publishes” info in the bottom of the window. Exceeding that limit will leave the Site stuck in the last publish until it resets.*

Session Wrap Up - 5 min

- Any questions about what we covered that we did not get to today?
- Any concerns?
- Schedule next session
 - If they are not ready to schedule, be sure to include a scheduler link in the follow up email
 - Reinforce any objections with the idea that it's best to review the material together with a consistent schedule so they get the most out of their sessions



Thank them for their time!

Admin Wrap Up - 20 min

- Log time in Tempo
- Send follow up email to merchant

 SS&M Standalone

STORE SETUP & MANAGEMENT

Module | Design - Catalyst / Makeswift

Facilitator Guide



Note: Please update this FG as needed and have a Lead Trainer approve your changes. If there are significant changes and you are concerned about formatting, please submit a Jira ticket if an ID review is needed.

© 2024 BigCommerce Pty. Ltd. All rights reserved.

No part of this publication may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission of the publisher, except in the case of brief quotations embodied in critical reviews and certain other noncommercial uses permitted by copyright law. For permission requests, write to BigCommerce Education at 11305 4 Points Dr. Austin, Texas 78726.

Version: 2.0 Updated: June 2024

NOTE: *The Store Setup & Management content and videos are regularly reviewed and updated, allowing you to stay current with the BigCommerce platform and your professional development.*

Table of Contents

Module | Design - Catalyst / Makeswift

- ❑ [**Lesson 1:** Introduction to Catalyst](#)
- ❑ [**Lesson 2:** Makeswift Visual Builder](#)

Module Prerequisites:




Participant Prerequisites:




- ☐ Have a sandbox store ready
- ☐ Read the following articles:
 - ☐ **KB Resource:** [Catalyst for Composable Commerce](#)
 - ☐ **KB Resource:** [Makeswift Visual Builder](#)
 - ☐ **KB Resource:** [Creating a Catalyst Storefront](#)

Facilitator Prerequisites:

- ☐ Have a sandbox store ready
- ☐ Has access to One-Click Catalyst configuration
- ☐ Has access to Makeswift

Materials Key:

Icon	Definition
	Facilitation Tip
	Discussion Question
	Lab

Icon	Definition
	Live/Demonstration
	Internal Notes
	External Notes

Lesson 1: Introduction to Catalyst

Learning Objectives:





- Understand the purpose of Catalyst in a Headless BigCommerce build
 - Set up Catalyst using the One-Click Catalyst functionality
-





Participant Prerequisites:

- ☐ Intend to use Catalyst to implement a headless store environment
- ☐ Learner must be [Store Owner](#) or have custom permissions to configure One-Click Catalyst

Facilitator Prerequisites:

- ☐ Reviewed the most current elearning: SS&M: Module Catalyst/Makeswift
- ☐

Topic	Actions (What do learners see?)	Script Highlights
Catalyst - Overview	 Show: <input type="checkbox"/> Channels	<ul style="list-style-type: none"> • Catalyst is a next generation design framework optimized with Next.js to improve performance, flexibility, scalability, and developer experience. In short, Catalyst is built to support Composable Commerce <ul style="list-style-type: none"> ◦ For non-technical users, it can be described as a codebase that provides highly optimized, reusable, and customizable components to improve both the design experience and the experience for customers once the site is built. • When considering whether Catalyst is right for you, the following are qualifying criteria: <ul style="list-style-type: none"> ◦ Willingness to manage design using some measure of code and technical knowledge, ◦ Primarily B2C – or B2B with Buyer Portal, ◦ Plan to use a composable/headless implementation for BigCommerce, and ◦ Ability to host Catalyst off of BigCommerce due to the nature of the framework. <p> External:</p> <ul style="list-style-type: none"> • There isn't much to show to learners during the overview section, but it may be valuable to pull up the Catalyst One-Click app page (pre-install), as it gives a nice overview and provides extra talking points. <p> Ask:</p> <p> Internal:</p> <ul style="list-style-type: none"> • The way BigCommerce uses Catalyst is primarily as a theme framework similar to the Stencil framework currently in use. Similarly, Makeswift serves as an analog for the Page Builder. Catalyst provides the basic “theme” in the form of Vibes – the Soul Vibe is analogous to Cornerstone – and Makeswift allows editing for elements that allow further customization. Some elements can be hard coded in Catalyst to disallow editing in Makeswift.

Topic	Actions (What do learners see?)	Script Highlights
Catalyst - One-Click Catalyst	 Show: <ul style="list-style-type: none"> <input type="checkbox"/> Channels <ul style="list-style-type: none"> <input type="checkbox"/> Create new channel <input type="checkbox"/> One-Click Catalyst 	<ul style="list-style-type: none"> • In order to get merchants started using Catalyst efficiently, BigCommerce provides a quick-deployment tool called One-Click Catalyst. This tool automatically configures a new storefront to work with Catalyst using Makeswift as a front-end design tool similar to the Page Builder, which may be used for the cart, checkout, and order confirmation pages in a headless build. <ul style="list-style-type: none"> ◦ Even using One-Click Catalyst, merchants will likely need to do some measure of code management unless they plan to use the default Vercel hosting, which is strongly discouraged, as it is intended to be a sandbox only. ◦ The One-Click Catalyst app offers storefront management specific to those using Catalyst, and storefronts can be added, removed, or edited as needed. ◦ In the actions menu within One-Click Catalyst is a Redeploy option that will update the deployment to the current build. • Current functionality of One-Click Catalyst does not fully enable Multi-Storefront, so those features should be enabled before using the tool. <ul style="list-style-type: none"> ◦ Navigate to Channels > Create Channel > Enable Storefront Seats, and enable all Multi-Storefront features. <p> External:</p> <ul style="list-style-type: none"> • Learners will need access to a multi-storefront control panel with at least one available seat in order for One-Click Catalyst to configure a channel, deploy Catalyst, and use Makeswift. If a merchant is planning to set up their own Catalyst deployment, their developer will need to assist. <p> Ask:</p> <p> Internal:</p> <ul style="list-style-type: none"> • While it is possible to use the default hosting, this should be considered only a temporary option for testing and familiarization.

References

- **KB Resource:** [Creating a Catalyst Storefront](#)
- **KB Resource:** [Catalyst for Composable Commerce](#)

Lesson 2: Makeswift

Learning Objectives:





- Access Makeswift from BigCommerce
 - Identify the primary features of Makeswift used for customising a headless storefront
 - Add and update pages in Makeswift using Components and the Properties Sidebar
 - Understand the advantages and limitations of Makeswift in supporting the Catalyst framework for headless commerce
-





Participant Prerequisites:





- ☐ Have set up One-Click Catalyst using sample data
- ☐ Learner must be [System Admin](#) or have custom permissions to access Channels
- ☐ Learner must be a user in the Makeswift workspace associated with the store.





Facilitator Prerequisites:





- ☐ Reviewed the most current elearning: SS&M: Module Catalyst/Makeswift
- ☐





Topic	Actions (What do learners see?)	Script Highlights
Makeswift Visual Builder	 Show: <ul style="list-style-type: none"> <input type="checkbox"/> Channels (or Apps) <input type="checkbox"/> One-Click Catalyst <input type="checkbox"/> Makeswift 	<ul style="list-style-type: none"> • The Makeswift Visual Builder is a rapid design front-end built to work with a variety of design frameworks such as, but not limited to Catalyst. By using the One-Click Catalyst configuration, a storefront is automatically set up and configured to be built using the Makeswift builder. Some call-outs about this include <ul style="list-style-type: none"> ○ Access - Makeswift is accessible from the control panel from either Channels or Apps and opening One-Click Catalyst then selecting the correct storefront once there. ○ Usability - The site builder is designed with simplicity and ease of use in mind, but it's a robust and complete site design front-end allowing for minimal use of code while not prohibiting it if needed. ○ Purpose - For storefronts created via One-Click Catalyst, Makeswift replaces the Stencil Page Builder as a visual editor.  External: <ul style="list-style-type: none"> • Merchants will likely need to configure a Makeswift account unless they have previously used the tool. This process should present itself upon clicking the Edit in Makeswift button and can be done using the account link options provided.  Ask:  Internal: <ul style="list-style-type: none"> • Participants will likely need to configure a Makeswift account unless they have previously used the tool. This process should present itself upon clicking the Edit in Makeswift button and can be done using the account link options provided.





Topic	Actions (What do learners see?)	Script Highlights
Makeswift Visual Builder - Orientation	 Show: <ul style="list-style-type: none"> <input type="checkbox"/> Control Bar <input type="checkbox"/> Navigation Sidebar <input type="checkbox"/> Components Toolbar <input type="checkbox"/> Canvas <input type="checkbox"/> Properties Sidebar 	<ul style="list-style-type: none"> • The Makeswift workspace is home to one or more sites available to design on a given account. All design and navigation features of Makeswift are available from a single workspace, including Sites, Pages, Files, Data, and the functions needed to customize them. <ul style="list-style-type: none"> ○ Control Bar - Contains navigation, preview, and site management functions, including Publish, which is how changes are pushed to the storefront. ○ Navigation Sidebar - Gives access to settings, pages, data, and other sites within the workspace ○ Components Toolbar - Design elements such as text, images, and more complex features can be pulled from this area by drag-and-drop interface with the Canvas. Also includes the controls for Build and Interact modes. ○ Canvas - Serves as both preview and working area to adjust the design of the active page with a click-and-drag interface. ○ Properties Sidebar - Provides customization features for the active page, element, or component.  External: <ul style="list-style-type: none"> • Show merchants the various components around the site builder. For each section of the workspace, it is helpful to click into one or more of the functions both to draw attention and to solidify the relationship with the content.  Ask:  Internal: <ul style="list-style-type: none"> • When using the Makeswift builder, best practice for navigating between pages is to use the URL bar, as navigating in Interact Mode can cause some unexpected behavior. The Navigation Sidebar is the next best choice.





Topic	Actions (What do learners see?)	Script Highlights
Makeswift Visual Builder - Settings	 Show: <ul style="list-style-type: none"> <input type="checkbox"/> Settings Menu <input type="checkbox"/> Site Options <ul style="list-style-type: none"> <input type="checkbox"/> Site <input type="checkbox"/> Host <input type="checkbox"/> Domains <input type="checkbox"/> Workspace Options <ul style="list-style-type: none"> <input type="checkbox"/> Users <input type="checkbox"/> Plans <input type="checkbox"/> Billing <input type="checkbox"/> Integrations 	<ul style="list-style-type: none"> • The Settings menu is available in the Navigation Sidebar and provides access to several important functions for managing and maintaining a site and a workspace overall. <ul style="list-style-type: none"> ○ Site > Site - Site name, favicon, and sitemap are configured here, as is the ability to archive or delete a site from the account. ○ Site > Host - Because Catalyst and Makeswift are part of a headless implementation, this allows adjustment of where the relevant codebase is stored. ○ Site > Domains - Domain management is primarily managed from the control panel, but secondary domains may be added here to ensure correct content delivery. ○ Workspace > Users - Multiple users can be added to the workspace with various permission levels based on needs. User seats are limited by plan, but Guests are not counted against this limit. ○ Workspace > Plans/Billing - The plans page provides an overview of what features are locked and at what level, while the billing page gives an overview of financial details attached to the account and a payment method can be added as needed. ○ Workspace > Integrations - Lists currently implemented integrations and provides access to manage them as necessary. <p> External:</p> <ul style="list-style-type: none"> • This is an excellent opportunity to make the point that extended functionality in Makeswift is plan locked, and merchants may need to upgrade if their team needs those extended features. <p> Ask:</p> <p> Internal:</p> <ul style="list-style-type: none"> •





Topic	Actions (What do learners see?)	Script Highlights
Makeswift Visual Builder - Navigation and Properties Sidebars	<p> Show:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Site Selector <input type="checkbox"/> Pages Tab <input type="checkbox"/> Elements Tab <ul style="list-style-type: none"> <input type="checkbox"/> Selecting components and subcomponents. <input type="checkbox"/> Design Tab <input type="checkbox"/> Files Menu 	<ul style="list-style-type: none"> • The navigation sidebar provides access to change the current active element and manage general design assets for the current site's theme. <ul style="list-style-type: none"> ○ Site Selector - For workspaces with more than one site, this can be used to navigate between them. Site data is local to the specific site, while workspace data is global to the workspace. ○ Pages Tab - Provides navigation space for static and unique pages for the purpose of design work. Clicking on a page sets it as the active element for editing. A grey dot on a page means it is not public. ○ Elements Tab - Provides navigation for on-page elements of all types. Clicking an element will focus it in the Canvas and set it as the active element for editing. ○ Design Tab - Lists all design assets for the current site design theme. These include but need not be limited to font configuration and colors. Assets listed here are linked and update all elements that reference them. ○ Files Menu - shows files that can be referenced from anywhere in Makeswift. Functionality is standard and similar to Google Drive <p> External:</p> <ul style="list-style-type: none"> • When selecting a page or an element, indicate the change of the Properties Sidebar, and show the features that are editable for the given object. When showing the design tab, it can be beneficial to have an element pulled up and show how its properties are linked to the elements being edited • In particular, demonstrate ease of use in selecting components and sub-components for editing from the Elements tab compared to the Canvas. <p> Ask:</p> <p> Internal:</p>

Topic	Actions (What do learners see?)	Script Highlights
<p>Makeswift Visual Builder - Box Components</p>	<p> Show:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Box Components <ul style="list-style-type: none"> <input type="checkbox"/> Position <input type="checkbox"/> Layout <input type="checkbox"/> Properties 	<ul style="list-style-type: none"> • Box components in particular serve as a versatile starting point for layout, design, and content that can be deeply nested as necessary to allow control of features and functionality within specific sections of the page. <ul style="list-style-type: none"> ◦ The default behavior for Boxes is to arrange content in a stack, but a section of a Box can be switched between stack and column layouts by clicking one of the dashed lines in the parent component. ◦ Each Box may contain one or more other components for displaying specific content, including other Boxes, arranged based on the rules of the Box they are in. ◦ Design choices (such as color) made for a Box affect the components nested inside it. <p> External:</p> <ul style="list-style-type: none"> • Demonstrate nesting and layout behavior by inserting some Boxes such that they are nested, then click the various dashed lines for clearer visuals on the comment made about them above. • When demonstrating the use of Boxes, also point out the dynamic behavior of the Elements Tab and Canvas as you hover over the individual components. <p> Ask:</p> <p> Internal:</p> <ul style="list-style-type: none"> • When comparing the Visual Builder to the Page Builder, Box components are a more advanced version of the native Layout widget, allowing more freedom for nesting and general design, while also serving as standalone content areas.

Topic	Actions (What do learners see?)	Script Highlights
Makeswift Visual Builder - Components Toolbar	 Show: <ul style="list-style-type: none"> <input type="checkbox"/> Drag-and-Drop functionality <input type="checkbox"/> Components library <input type="checkbox"/> Global Components <ul style="list-style-type: none"> <input type="checkbox"/> Detachment 	<ul style="list-style-type: none"> • Components are the primary design element of Makeswift. These are broken down as react components (box, text, image, button, and extended versions of them), global components, and Makeswift components. <ul style="list-style-type: none"> ◦ Built-in <i>components</i> - can be added to a page in any area not restricted by code configuration. These can then be customized as desired. This includes the basic components and the more advanced ones in the Component selector. ◦ Built-in <i>elements</i> - elements of the theme design such as the header, footer, and other elements that cannot be managed within Makeswift. These are almost entirely controlled through code. ◦ Global components are customized components that are saved for re-use around a site. Once a global component has been created, changes to any instance will carry through to the other instances <ul style="list-style-type: none"> ■ Global components can be detached from the rest of their instances and customized for a specific page or area. <p> External:</p> <ul style="list-style-type: none"> • Demonstrate the use of the Components toolbar to add Components, showing how to customize them with the Properties Sidebar and the click-based functionality in the Canvas. • Demonstrate converting a Component to Global. <ul style="list-style-type: none"> ◦ Name Global components with names like My/Global/Button to demonstrate grouping in the Component Selector. ◦ Demonstrate how to detach a Global component <p> Ask:</p> <p> Internal:</p> <ul style="list-style-type: none"> •

Topic	Actions (What do learners see?)	Script Highlights
Makeswift Visual Builder - Build and Interact	 Show: <ul style="list-style-type: none"> <input type="checkbox"/> Component Toolbar <ul style="list-style-type: none"> <input type="checkbox"/> Build <input type="checkbox"/> Interact 	<ul style="list-style-type: none"> • In order to verify functionality, conformity to style standards, and other details about site behavior, a user can switch into Interact mode, which allows the use of buttons, menus, and other interactive features. On the other hand, Build Mode allows the adjustment and customization of the active page and the active element generally. <ul style="list-style-type: none"> ◦ The Form-Factor selector is still available in Interact Mode to see how form-factor changes the impact of the design. ◦ Using the Navigation Sidebar or the URL tool in the Control Bar to navigate to a new page will automatically exit Interact Mode. ◦ Elements can still be customized in the Properties Sidebar in Interact Mode by selecting them from the Elements Tab in the Navigation Sidebar <p> External:</p> <ul style="list-style-type: none"> • Note that navigating to other pages in Interact Mode does not update the active page, so changes to other pages will not commit. A better choice would be to navigate to the new page in Build Mode – or use the URL bar to navigate, which will automatically switch to Build Mode – then switch back to Interact Mode. <p> Ask:</p> <p> Internal:</p> <ul style="list-style-type: none"> •

Topic	Actions (What do learners see?)	Script Highlights
Makeswift Visual Builder - Form-Factor	 Show: <ul style="list-style-type: none"> <input type="checkbox"/> Control Bar <ul style="list-style-type: none"> <input type="checkbox"/> Form-Factor selector <input type="checkbox"/> Form-Factor specific design <ul style="list-style-type: none"> <input type="checkbox"/> Hide an element in Large View <input type="checkbox"/> Switch between Large, Desktop, and Small to show propagation 	<ul style="list-style-type: none"> • Makeswift offers responsive design consideration directly in the workspace without having to open separate views or customization features. The form-factor dropdown allows the user to view the site as it would be delivered to a customer based on the profile of the browser they are in. • Switching between form-factors automatically updates the page to display the responsive design. • Editing the design of a specific element (not the content) in a form-factor will propagate to all the form-factors below it in the list. Meanwhile changes to lower form-factors do not propagate upward. <ul style="list-style-type: none"> ◦ Changing Desktop changes all others. ◦ Changing Large changes Medium and Small, but not XL or Desktop. • Demonstrate the feature by: <ul style="list-style-type: none"> ◦ Adding an element to the page and switching to Large form-factor. ◦ Adjusting the size and/or visibility of the element in the Large view. ◦ Switching between Large, Small, and Desktop to show how propagation carries downstream but not up.  External: <ul style="list-style-type: none"> • The default Catalyst/Makeswift implementation provides five form-factors, but more can be added via code if a merchant deems it necessary for their solution.  Ask:  Internal: <ul style="list-style-type: none"> •

Topic	Actions (What do learners see?)	Script Highlights
Makeswift Visual Builder - Publishing	 Show: <ul style="list-style-type: none"> <input type="checkbox"/> Publish Menu <ul style="list-style-type: none"> <input type="checkbox"/> Changes <input type="checkbox"/> Scheduled <input type="checkbox"/> History 	<ul style="list-style-type: none"> • Changes made in Makeswift are saved automatically. This means no extra <i>Save</i> step is required to update the design in general. • Changes are not delivered to customers until they are published. <ul style="list-style-type: none"> ○ Publish events can be pushed for any feature that has been added or updated since the last Publish event. This includes design assets, components, pages, and other features that are managed in the workspace. ○ On paid accounts, a Publish can be scheduled in advance, depending on the plan limits. ○ On paid accounts, Publish history is available for a certain time, depending on plan limits. <p> External:</p> <ul style="list-style-type: none"> • Note while discussing Publishing that publishes are limited to a certain number per month per account, not per site. <p> Ask:</p> <p> Internal:</p> <ul style="list-style-type: none"> •

References

- **KB Resource:** [Makeswift Visual Builder](#)
- **KB Resource:** [Catalyst Components in Makeswift](#)
- **Makeswift Docs Resource:** [Makeswift Quickstart Guide](#)