

# Content modeling: practice makes perfect

How Clover's content infrastructure works for both editors and developer

October 9, 2019

1



Time frame:

Amelia - Introductions, Agenda, Content modeling overview - 5 - 7 mins

Rachel - Projects review 30 - 35 mins (10 -12 mins/project)

Amelia - Wrap up & Summary of Key Learning - 5 mins

Rachel & Amelia - Audience Q&A - 15 mins

# Housekeeping

Before we dive in...

- This webinar is being recorded; we will share the recording and slide deck with all registrants shortly -- keep an eye on your inbox!
- Please add your questions in the Q&A box, we will answer them after the presentation during the live Q&A session at the end
- To get in touch, write to us at [team@contentfulmail.com](mailto:team@contentfulmail.com)
- Join the conversation on Twitter with [#CTFwebinar](#)



# Introductions



Rachel Church  
Sr Software Engineer



Amelia Winger-Bearskin  
Developer Evangelist

3



**Amelia:** I'm Amelia Winger-Bearskin, Developer Evangelist at Contentful.  
I'm here with Rachel Church

**Rachel:** "Hello everyone. My name is Rachel Church and I am a senior software engineer at Clover Network. I have been using Contentful for the last two years for a number of different projects here at Clover, and I am excited to share my experience with you."

# Agenda

1. Content modeling -  
what are we talking about?
2. Three Projects:  
Public Site, Help Center, Marketing Site
  - Business Drivers
  - Process + Example
  - Challenges
3. Key learnings

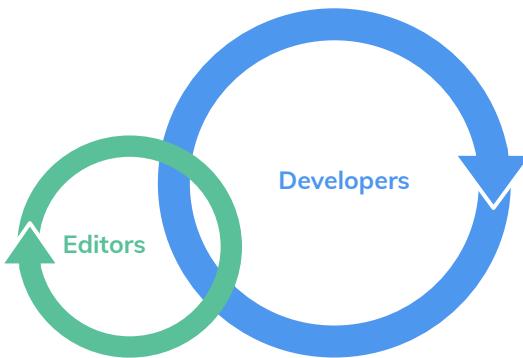


# Digital teams (devs and editors) want to move fast

Both with and without each other

## Create & Manage Content

- Create landing pages
- Localize a microsite
- Launch a campaign
- Run A/B tests
- Update homepage promos
- Tweak signup page copy



## Build & Deliver Features

- Launch a new app
- Integrate with a DAM
- Iterate a feature
- Update a design system
- Update a content model
- Test/QA new features
- Import legacy content
- Ship bug fixes

5



A:

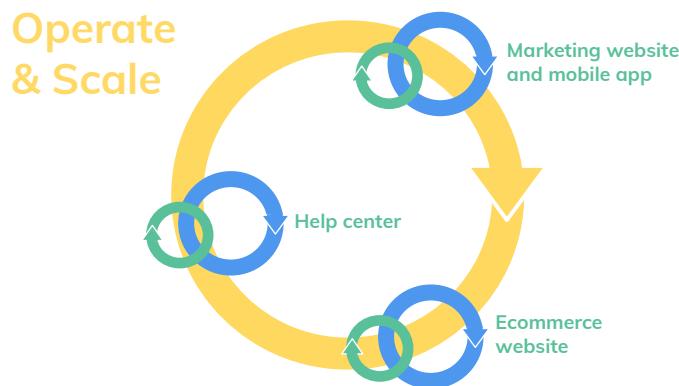
- Our webinar today is all about content modeling and how to optimize your content models for cross functional teams.
- Building digital products (like a website) takes cross functional teams.
- Content marketers creating & managing content; they build campaigns. Optimize conversion on landing pages; are mindful of SEO in the content and order of their pages.

Developers, like me and Rachel, focus on building & delivering with content models that meet the needs of editors using the Contentful WebApp and making sure integrations work flawlessly for cross-functional teams.

Fact of life: Teams have different priorities and pressures - Communication is key.

# Content platforms create agility at scale

Reusable architecture & processes allow teams to move fast across an organization



6



A:

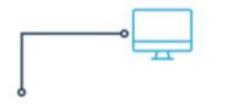
- For teams to scale with reusable content workflows need to happen in parallel.
- To do this effectively, developers need to build content models that are optimized for those writing and updating the content - the marketers.
- We should be doing what we do best - delivering amazing solutions with code, so that our marketing teams can do what they do best, getting our company's story out into the world in the most effective way possible.
- in order to build an elegant solution you'll need empathy. **Feel the pains of your developer or marketing counterpart! It will help with content modeling, we promise.**
- And listening building and iterating helps your company move faster!

# Basic Contentful Architecture

Developers work in code...



All content goes to  
the same place...



Authors work in the web app...



...and is delivered  
to any platform!



7



A: content infrastructure provides structured content that can be reusable - enabling devs and editors to work in parallel and move faster - this diagram gives a zoomed out, simplified view of how Contentful makes this possible.

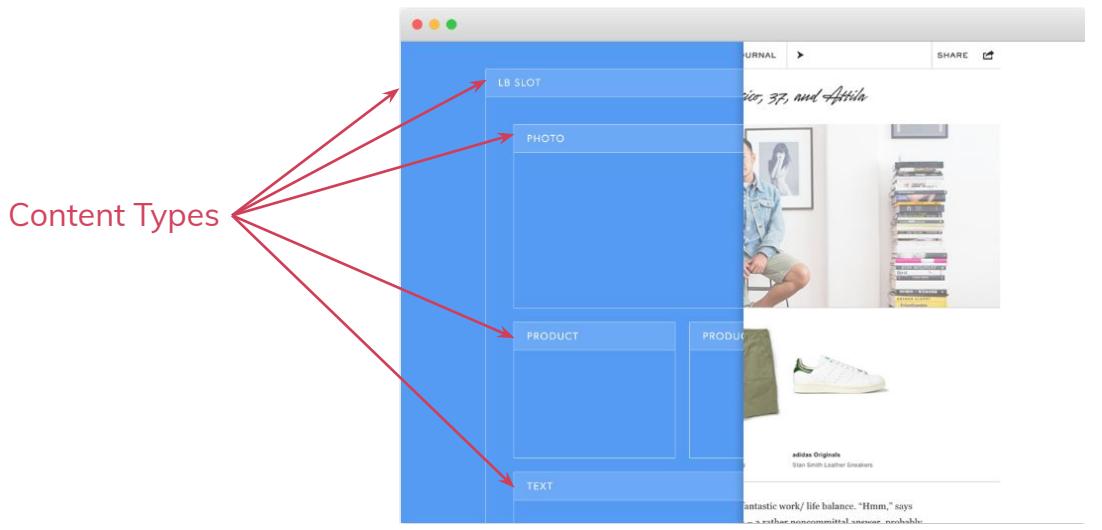
## Basic Contentful Architecture

Developers work in code

Authors work in the Contentful Web app

all content goes to the same place, and is deployed where the user is and meets them where your story needs to be taken. It could be a website, mobile app, smart watch, voice activated navigation in a smart home, ajumbotron!

# Reusable content means “structured” content



8



Content Modeling: you have to think about content a bit differently than you may have in the past.

A web page is made out of building blocks. Breaking a “blob” of content into smaller components is what makes it possible to reuse and share content easily. Contentful calls these building blocks “**content types**.”

Just like structured data, Contentful is helping you structure content. Identify chunks of content that could be used in multiple places - store it in one structured infrastructure, it is easier to deploy it again and again across different apps and outputs (like big screens, in-store screens or pocket-sized screens).

It is also important to identify reusable visual components so that the code for displaying that visual representation (think: left aligned, or hero image) is maintained in one place.

Model once and publish everywhere is a common phrase, however sometimes you don’t get it right the first time, which is why practice makes perfect! Sometimes that practice is also learning about automating changes as well (with migration scripts!).

# Contentful Content Modeling Methodology

STEP 1

STEP 2

STEP 3

STEP 4

STEP 5

STEP 6

## Test with Content Creators

- Watch authors use the content model (usability test)
- Create documentation, help and job aids

STEP 7

## Test with Developers

- Have developers QA the model against front-end, migration scripts, etc.

STEP 8

## Test with Designers

- Have designers test the content model by creating a variety of designs

9

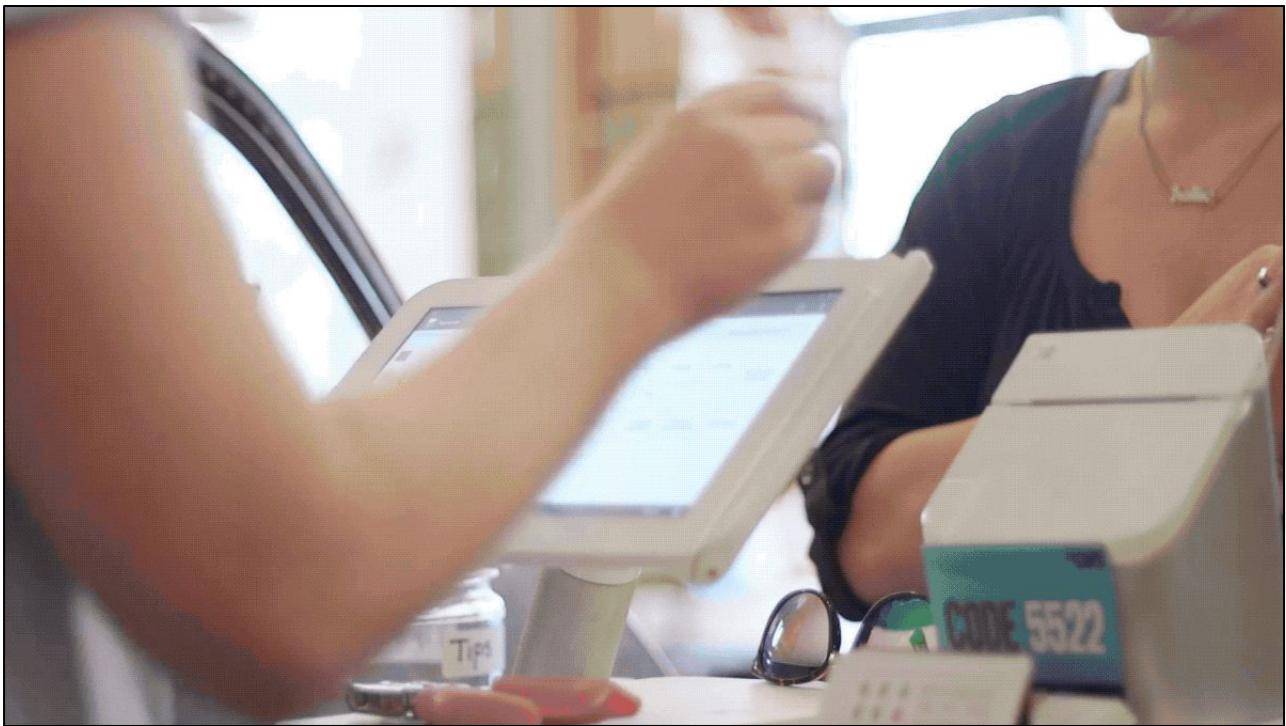
A:

We won't dive too deeply into this slide, it would be great to read this again after the webinar - for you to use in your content modeling projects this is something we developed to help your team's strategy with Content Modeling.

Rachel's team at Clover started this process, and like anything, practice makes perfect!

It is important to perfect your practice and make it work for you! A good planning strategy around your content will help your team, but so will a flexible and open attitude. Rachel will speak to some examples where it is important to listen to all stakeholders to make sure your content model works for all.

The short story: feedback is key; important for devs not to make assumptions on behalf of your content marketers. Test, revise, validate, test again! ----On that note, I'm going to hand this over to Rachel to bring you examples of Content Modeling in action. Handing this over to you Rachel!



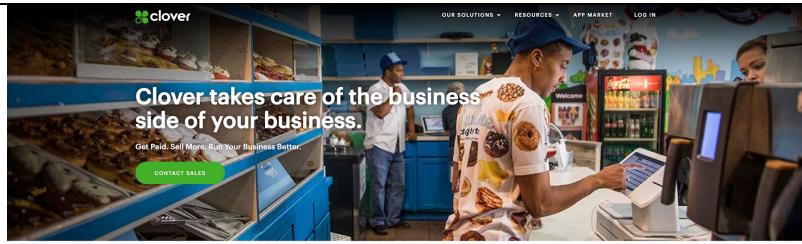
**Amelia:** Rachel tell us about Clover

**Rachel:**

- Cloud-based Android POS platform
- Manufacture our hardware
- Shipped over 1 million devices to shops, restaurants, and stadiums all over the world
- Empower merchants to run their business more easily and profitably
  - Provide a number of features and apps available through our app market

All three of the different web apps I will be going over are public so please feel free to visit them and click around.

I can try and answer any questions you have about them at the end.



Get Paid. Sell More. Run Your Business Better.



Clover Flex



Clover Mini



Clover Station



POS Software

Versatility at the countertop, in the aisle, or on the go.

Accept swipe, EMV chip, and NFC payments (like Apple Pay) right out of the box.

Our all-in-one POS solution boasts enhanced features, including fully integrated EMV<sup>®</sup> payment acceptance.

Intuitive software for all your business needs.



### Rachel:

- This is a screenshot of Clover's homepage. Our homepage was Clover's first project using Contentful
- Initial development occurred before I joined Clover and when there was still a small development team

Amelia: What were the business drivers for this project?

### Rachel:

- Give content/marketing team control to edit content and create new pages
- Faster to make changes to the homepage

Amelia: Who built the content model?

Rachel: Someone from the design team took the lead on comparing CMS systems and setting up the content model.

- Since he is on the design team and is used to thinking of things in terms of UI, the content model is very UI driven

*Transition to next slide:*

- Let me break down this screenshot we are looking at...

The screenshot shows the Clover website homepage. At the top, there's a navigation bar with links for 'OUR SOLUTIONS', 'RESOURCES', 'API MARKET', and 'LOG IN'. Below the navigation is a 'Hero Header' with the text 'Clover takes care of the business side of your business.' and a 'Hero Subheading' with 'Get Paid. Sell More. Run Your Business Better.'. To the right of the subheading is a 'Hero Button Text' ('CONTACT SALES') and a 'Hero Button URL'. A 'Hero Background Image' shows a person working at a cashier counter. Below this is a 'Full Width Section' containing the tagline 'Get Paid. Sell More. Run Your Business Better.'.

Further down is another 'Full Width Section' featuring four product cards:

- Clover Flex**: Versatility at the countertop, in the aisle, or on the go.
- Clover Mini**: Accept swipe, EMV chip, and NFC payments (like Apple Pay) right out of the box.
- Clover Station**: Our all-in-one POS solution boasts enhanced features, including fully integrated EMV® payment acceptance.
- POS Software**: Intuitive software for all your business needs.

12



## Rachel:

- Homepage is represented by a 'Page' content type
  - Consists of fields to define the 'Hero'
  - Dynamic number of sections
    - Content creators have control over the sections
    - Content differs for each country and language
  - Meta information about the page
- The homepage is built using a 'Page - Full Width Sections' content type
  - Two other 'Page' types
- Other building blocks such as Row, Form, Tile make up different Sections

### *Transition to next slide:*

So this is a high-level overview of the content-modeling strategy we used, let me dive deeper into the more technical side of how this app is built..

# Technology stack



The screenshot shows a Contentful interface for creating a new route. The route is named 'Clover Home Page'. The 'Route Description' field contains 'Clover Home Page' and has a character limit of 16 characters, with a note that it requires less than 256 characters. The 'Route URLs' field is empty and prompts the user to 'Type the value and hit enter'. The 'Content Depth' dropdown is set to 4. In the 'Page Reference' section, the page 'Page - Full-Width Sections' is selected, and the status is 'PUBLISHED'. There is also a 'Welcome to Clover' entry.

13



## Rachel:

- Use EmberJS as our web framework.
- Code lives in mono-repo like repository with directories for each app hosted on the clover.com domain
- Url routes are statically defined within the code.
  - Means the code knows beforehand what to show when a user hits a specific URL.
- Clover homepage is completely defined in Contentful so new pages can be created on the fly.
- If the user goes to a URL that is not statically defined we send a request to Contentful to check if there is a 'Route' defined that matches
  - If an entry is returned we send another request to get the page's data

## *Transition to next slide:*

I think this is a pretty cool concept so I want to spend a little bit of time looking at the code that handles this.

For the non-technical audience, don't worry I will go back to the high-level content-modeling strategy shortly.

```
getPageForUrl(locale, url) {
  return this.getRouteForUrl(locale, url).then(route => {
    let pageId = route.pageReference.sys.id;

    return this.getClient().getEntries({
      'sys.id': pageId,
      include: route.contentDepth || this.DEFAULT_CONTENT_DEPTH,
      locale: this.formatContentfulLocale(locale)
    }).then(entry => ({
      route,
      entry: entry.items[0]
    }));
  });
},
```

14



This is actual code copy+pasted from Clover's production Ember code.

This function is called anytime a static URL route was not matched so we can determine if it is a valid dynamic route.

```
getRouteForUrl(locale, url) {
  if (!this.cache.routes[url]) {
    this.cache.routes[url] = this.getClient().getEntries({
      content_type: 'route',
      'fields.routeUrl': url,
      locale: this.formatContentfulLocale(locale)
    }).then(routes => routes.items[0].fields);
  }

  return this.cache.routes[url];
},
```

*contentful.createClient*  
<https://contentful.github.io/contentful.js/contentful/7.10.0/contentful.html#.createClient>



```
getPageForUrl(locale, url) {
  return this.getRouteForUrl(locale, url).then(route => {
    let pageId = route.pageReference.sys.id;

    return this.getClient().getEntries({
      'sys.id': pageId,
      include: route.contentDepth || this.DEFAULT_CONTENT_DEPTH,
      locale: this.formatContentfulLocale(locale)
    }).then(entry => ({
      route,
      entry: entry.items[0]
    }));
  });
},
```

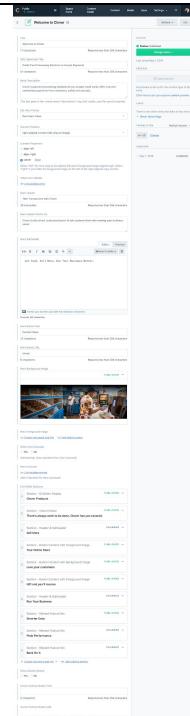
16



*Transition to next slide:* Now let's go back to the Contentful UI and take a look at the Page content type that we just fetched



17



That's a  
long  
content  
model



## CHALLENGES/PROS/CONS

Now let's dive into what makes up our page content type..

- 3 different 'Page' types.
- There are 24 different 'Section' types that can be used on a page.
- There are a total of 48 different content models.
- Looking at the number of different content types is overwhelming and no one truly understands what each field controls without looking at the code.

# What we learned

18



Avoid having new content models for variations of the same thing

What is clear to the person setting up the content models may not be clear to everyone else

## LESSONS LEARNED

- Re-use content types whenever possible.
  - Avoid having new content models for variations of the same thing
  - '**Section - Offset Columns**' vs '**Section - Columns**' could have true/false flags instead
- What is clear to the person setting up the models may not be clear to everyone else
  - Add descriptions to every model and every field
  - Add strict validation whenever possible. Often easier to loosen validation than to make more strict
  - Always go through a model review with all stakeholders

### Rachel:

- This is a screenshot of our second project using Contentful - Clover's help site
- Help Center was previously in Wordpress
- Clover was expanding into more countries and localization and filtering content became very important
  - (click to show french) This is what the site looks like for Canada in French

**Amelia:** How did the needs of this project differ than the Clover Homepage?

### Rachel:

- The same desire to allow the content team to drive content changes and create new pages themselves existed
- Want users to be able to do a Google search and find an answer on our help site. Optimized for Search Engines.
- Emphasis on localization and showing/hiding content based on the user's country

**Amelia:** Who built this content model?

**Rachel:**

- I worked side-by-side with the head of the content team to build the content types.
  - Process took weeks and a lot of trial/error as we built the app
- Built 16 different content models

*Transition to next slide:*

- Let me breakdown the different content models we created..

## CONTENT MODEL EXPLANATION (1/3)

- Four different page types: Homepage, Category, Guide, and Article
  - Different homepage content for different countries
- Each page has a title, pathname, and fields to add content
- Original debate was to have one 'Page' type with a dropdown to select the page type, vs having different models for each

The screenshot shows the Clover Help Center homepage with several article categories displayed:

- Set Up Your System** (Category):
  - TITLE: String**: New to Clover? Learn how to set up your Clover hardware and software. And hit the ground running.
  - INTRO: String**: New to Clover? Learn how to set up your Clover hardware and software. And hit the ground running.
  - Links: GUIDE/Article List**:
    - Clover Station 2018
    - Clover Station
    - Clover Mini[View More](#)
- Run Your Business** (Category):
  - TITLE: String**: Learn how to use Clover to run your business.
  - INTRO: String**: Learn how to use Clover to run your business.
  - Links: Article List**:
    - Bank deposits: what you need to know
    - Orders and payments
    - Refunds, returns, and exchanges[View More](#)
- Recommended Articles** (Section):
  - ARTICLE** (Category):
    - Set up payments
    - Set up an order printer
    - Set up order receipts
    - Take a payment
    - Close out the current batch
    - Run reports on the Web Dashboard
  - TITLE: String**: Set up payments
- Troubleshooting** (Category):
  - TITLE: String**: Find a solution to the issue you're having. Or get answers to a question.
  - INTRO: String**: Find a solution to the issue you're having. Or get answers to a question.
- Apps: Tools for Your Business** (Category):
  - TITLE: String**: Learn how to use Clover's core features. Need more tools for running your business? Find apps in the Clover App Market.
  - INTRO: String**: Learn how to use Clover's core features. Need more tools for running your business? Find apps in the Clover App Market.
- Your Clover Account** (Category):
  - TITLE: String**: Learn about your Clover account and how to manage access and billing.
  - INTRO: String**: Learn about your Clover account and how to manage access and billing.
- Clover Web Dashboard** (Category):
  - TITLE: String**: Manage your business while you're away from your Clover device. It's easy using the Clover Web Dashboard.
  - INTRO: String**: Manage your business while you're away from your Clover device. It's easy using the Clover Web Dashboard.
- Protect Your Clover System** (Category):
  - TITLE: String**: Learn ways to protect your Clover system against unauthorized access and to handle situations such as stolen or tampered devices.
  - INTRO: String**: Learn ways to protect your Clover system against unauthorized access and to handle situations such as stolen or tampered devices.
- Privacy & Security** (Category):
  - TITLE: String**: Learn how Clover secures and protects financial data on your Clover system, and preserves privacy for your customers.
  - INTRO: String**: Learn how Clover secures and protects financial data on your Clover system, and preserves privacy for your customers.

## CONTENT MODEL EXPLANATION (2/3)

Can see how some of the fields of the referenced entries are used on the homepage.



## Technology stack



24



### Rachel:

- The app was written in React and is deployed as a static app.
  - This means the site's content will remain the same until the app is rebuilt and redeployed.
  - Differs from the way our homepage is built where as soon as something is published in Contentful it can be seen live in production.
- We use an npm package called 'react-static' which is a part of both the build process and the runtime
  - Similar to Gatsby

Amelia: What drove the decision to make the help app static?

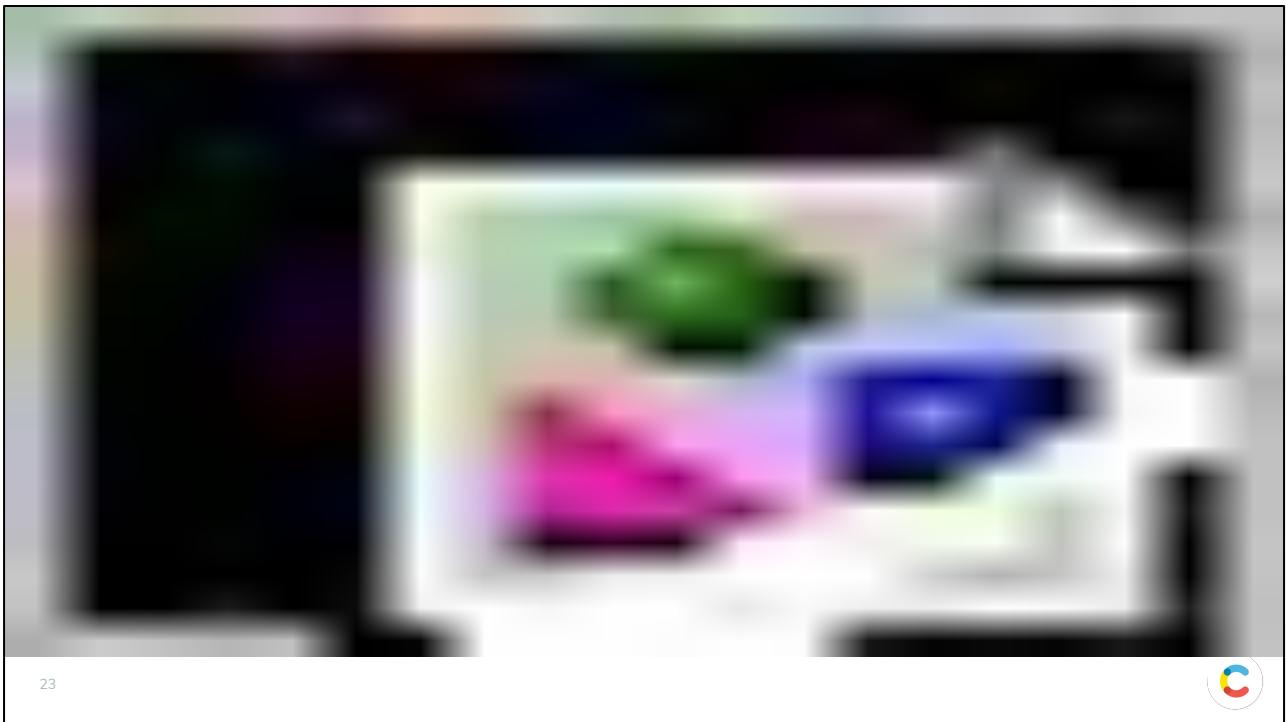
### Rachel:

- Important to us to have the help site be optimized for search engines
  - Static sites make this easy because the routes and data are determined upfront
  - react-static generates a sitemap file that crawlers can use to index the site's content
- Also makes the site a little more stable
  - Content changes are treated similar to code changes

- Any new build includes a snapshot of the Contentful data and can be deployed to our dev and stage environments for testing before being deployed to production.
- The snapshot ensured that after going through QA and verification the content deployed to production is stable.

*Transition to next slide:*

Now let me show you something cool about static site generators like react-static and Gatsby



23



Hopefully any web devs watching this should recognize the chrome dev-tools with the network tab open at the bottom.

react-static and Gatsby use intersection observers to preload the pages linked to from the page when they appear in the viewport.

If the user were to click on any of these links it would make them feel like they loaded instantly because the data was already cached.

The video will loop over again and watch as I scroll down how a request is made for the 'take a payment' page.

Static site generators use server-side rendering

- This means the pages and content are built out by the server so they can be sent to the browser ready to go

**Title**

## Make a full refund

**Devices**



There are three apps you can use to make a full refund:

**Intro**

- To refund an entire order, use the Orders app. The Orders app is only available on the Register Lite Plan, Register Plan, and Classic Plan.
- To manually refund an amount that's not tied to a transaction, use the Refund app.
- To refund a specific transaction, use the Transactions app to both find it and make the refund.
- To refund transactions made on the web or device, use the Transactions app.

**Section Title**

### Refund an order

When you create and take payment for an order in the Register app, the order is recorded in the Orders app. You can see how many items were in the order and how many payments were made.

When you refund an entire order, you refund the total cost of all items plus any taxes, flat fees, and additional charges.

To refund an entire order:

- 1 Open the Orders app.
- 2 Find and tap the order you want to refund.
- 3 Tap Exchange/Refund.
- 4 Tap Refund Payment on the popup window.
- 5 Tap the payment you want to refund.
- 6 Tap Refund.
- 7 Tap REFUND in the dialog box.
- 8 Swipe or insert the card or manually enter the card number.

In this article

Refund an order  
Issue a manual refund  
Refund a transaction

**TOC**

**Rachel:**

- This screenshot is of one of the Article pages - the meat and potatoes of the site where the majority of the content lives
- Take a second to look at all the different blocks - how would you break this page up?
  - Article **title**
  - list of **Clover devices** that this article applies to
  - Intro paragraph
  - **Table of contents**
  - Section
    - Title
    - Body text
- What fields would be required and which would be optional?
- How do we allow content creators to add links and lists?
- Now the hard part - what if the content needs to change when different devices are selected?

# Make a full refund

Web Dashboard

Clover Station  
2018

Clover Station

Clover Mini

Clover Flex

Clover Mobile

In this article

[Issue a manual refund](#)  
**Refund a transaction**



There are three apps you can use to make a full refund:

- To refund an entire order, use the Orders app. The Orders app is only available on the Register Lite Plan, Register Plan, and Classic Plan.
- To manually refund an amount that's not tied to a transaction, use the Refund app.
- To refund a specific transaction, use the Transactions app to both find it and make the refund.
- To refund transactions made on the web or device, use the Transactions app.

## Issue a manual refund

If you need to refund a specific amount, such as half the amount paid in a single transaction, you can make a manual refund. Manual refunds are made in the Refund app by entering in the exact amount you want to refund.

All manual refunds will be recorded in the Transactions app.

To issue a manual refund:

- 1 Open the Refund app.
- 2 Enter a refund amount.
- 3 Tap **Issue Refund** and confirm.
- 4 Swipe a payment card, insert a chip card, tap for contactless payment, or manually enter the card number.

## Refund a transaction

Your Clover device considers each payment as a transaction. When refunding a transaction, the full

**Rachel:**

- This is the same article, but with content specific to the Clover mini device
- And to further complicate it, what if we need content to be different for other english speaking countries?

ARTICLE

# Make a full refund

**Title**

Web Dashboard    Clover Station 2018    Clover Station    **Clover Mini**    Clover Flex    Clover Mobile

In this article

- Issue a manual refund
- Refund a transaction

There are three apps you can use to make a full refund:

- To refund an entire order, use the Orders app. The Orders app is only available on the Register Lite Plan, Register Plan, and Classic Plan.
- To manually refund an amount that's not tied to a transaction, use the Refund app.
- To refund a specific transaction, use the Transactions app to both find it and make the refund.
- To refund transactions made on the web or device, use the Transactions app.

**Section**

## Issue a manual refund

**SectionTitle**

**Paragraph** If you need to refund a specific amount, such as half the amount paid in a single transaction, you can make a manual refund. Manual refunds are made in the Refund app by entering in the exact amount you want to refund.

All manual refunds will be recorded in the Transactions app.

To issue a manual refund:

- Open the Refund app.
- Enter a refund amount.
- Tap Issue Refund and confirm.
- Swipe a payment card, insert a chip card, tap for contactless payment, or manually enter the card number.

**Section**

## Refund a transaction

Your Clover device considers each payment as a transaction. When refunding a transaction, the full



**Rachel:** This is how we decided to model the Article page

We have a paragraph content type that can be tagged to apply to specific Clover devices, and/or specific countries.

The code then iterates over the paragraphs and determines which to show/hide and which devices the user should see apply to the content

**Amelia:** Were there any challenges you faced?

**Rachel:**

The nesting of the content made it difficult for the content creators to get an overview of the content within an Article.

- Hard to see which content would show/hide for different cases.
- Content team was used to Wordpress where they could see a draft version of their changes visually. Huge pain point for them.

The static site made this even harder because there was no easy way to quickly see changes within the app

- Have to wait for the next build/deploy cycle
- Temp solution was to setup a dev environment on their laptops so

- they could run the code

In addition, if there was content changes that needed to be made outside of the normal build/deploy cycle, it was harder to make these changes quickly.

# What we learned

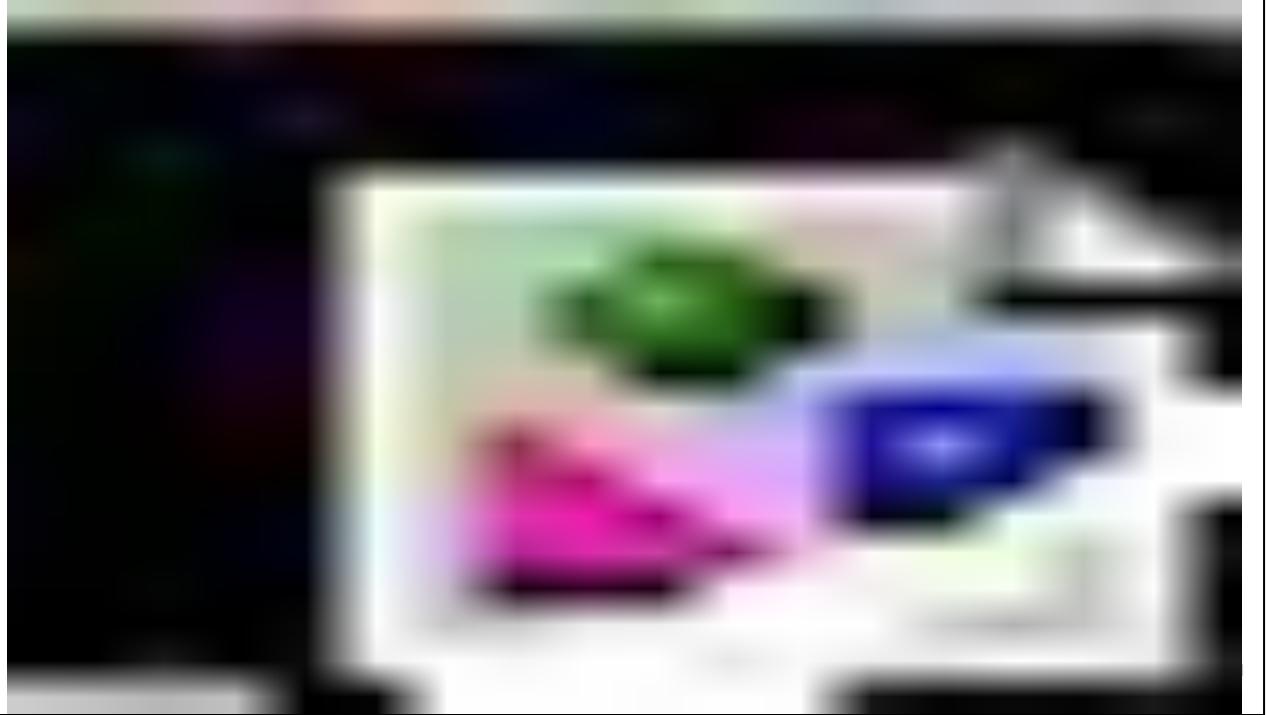
Content team **NEEDS** a quick and easy way to see their changes within the app's layout

The content model will most likely change



## LESSONS LEARNED

- Being able to visually see how all of the building blocks come together is very important especially when there are moving pieces.
- Regardless of how much planning you do you will likely have to adjust and add new fields to your model
  - Trial and error is a part of the content modeling process.
- What we do differently now:
  - Use Rich Text instead of markdown. This would reduce the need to the nesting of content within Section content types. The nesting makes it very hard for the content creators to see the big picture of the Article in one place.
  - Build UI Extensions as needed to make Contentful easier to use for the edge-cases we have
  - Make Help Center Dynamic instead of static
  - Not translate the permalinks of Help Center (over-complicated the flows)



**Rachel:**

This is Clover's new home page. We re-designed the UI and made a brand new Contentful space with fresh content models in 2018.

**Amelia:** What were the business drivers for this change?

**Rachel:**

Traditionally Clover has been sold through bank partners but in 2018 we added a direct sales channel through our website. With this came the need to redesign our homepage to better market our product and help potential merchants find the best fit solution.

We wanted to use the lessons we had learned from our last few projects to build a very clean and flexible content model.

This project had the same needs as the previous Clover homepage, but completely new content and design.

# Technology stack



Internal Name (required)  
Employee Management  
19 characters Requires less than 256 characters

Title (required)  
Employee Shift Planning & Work Schedule Maker | Clover  
54 characters Requires less than 256 characters

Slug (required)  
pos-systems/employee-management|  
The unique page name shown in the URL. Whitelisted urls are: pos/contact/promotion/event/campaign/pos-systems-software/pos-systems/solutions/pos-solutions/sitemap/resources/guides. Note that just '/' marks a page as the homepage.

Meta Description (required)  
Now you can easily manage employee work schedules and timesheets through our customized and fully integrated POS software solutions. Employee time tracking made simple.

Concise summary of the page's content.



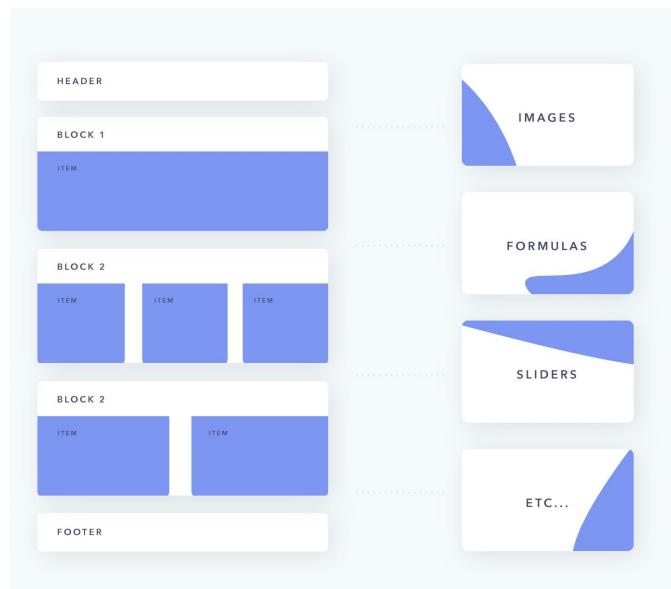
- The marketing site is fully dynamic.
- Routing works very similar to the original Clover Homepage

After reading an article on Contentful's blog on Topics and Assemblies we decided to follow this content modeling approach. The engineers took the site's designs and broke it up into building blocks.

Before I dive into how we built the content model, Amelia could you give a quick overview of Topics and Assemblies?

# Topics and Assemblies

More information [here](#)



- There are two primary objectives for using structured content. One is to identify chunks of content that could be reused in multiple places so that the content only has to be maintained in one place. The other is to identify reusable visual components so that the code for displaying that visual representation can be maintained in one place.
- We refer to content types intended primarily to control how content is displayed as “assemblies.” However, we have found that it is easier for some customers to refer to assemblies as modules (like Clover does) or components.

# Topics and Assemblies

More information [here](#)

Image Carousel



- One common example of an assembly is an image carousel. If you create an image carousel assembly, content authors can create multiple instances of it using different photos for each instance and your front-end code can render the carousel appropriately for each device or context. Note that assemblies are frequently nested. Assemblies can contain other assemblies, topics, and reference fields.
- Typically, one of the first steps in content modeling is identifying reusable content and assemblies. There are two strategies for this:
- Modeling content by focusing on the relationships inherent in the content itself
- Starting with a display mockup and identifying the content model required to build that display (while also thinking ahead to future needs)
- Either of these approaches can lead to a good content model, depending on the skills and preferences of those involved. In practice, content modeling often involves bouncing back and forth between these two perspectives.
-

Refer back to this article for more information

<https://www.contentful.com/r/knowledgebase/topics-and-assemblies/> (#1 organic result on Google if you search Topics and Assemblies)

**Amelia: [HAND THIS BACK TO RACHEL]** - How did you apply topics and assemblies to your content model for this project?

The screenshot shows the Clover Content Model Editor interface. On the left, there's a preview of a page with text and images about Clover POS. On the right, a 'Subject' assembly is being edited. It has a 'Fields (7)' tab selected, showing the following fields:

- Internal Name**: Short text
- Title**: Short text
- Image**: Media
- Video**: Media
- Blurb**: Long text
- Link**: Reference

Each field has a 'Settings' button to its right. Below the fields, there's a preview area showing a Clover device.

## CONTENT MODEL EXPLANATION

We called our main ‘Topic’ a ‘Subject’. It is the basic building block of the entire site and can be re-used in any Assembly to be displayed in a different layout/format.

A ‘Subject’ consists of a title, blurb, image, and link.

- Only the ‘title’ is required
- All entries have an ‘Internal Name’ field so it is easy for the content team to tag and search for content.

We have around 13 different ‘Assembly’ types which we call ‘Modules’. Each assembly just wraps one or more Subjects and defines how the content should be shown.

Goal was to allow the content to be completely re-usable in any of the different layouts

33

-- Deep dive into content model --

## CHALLENGES/PROS/CONS

- What we found was that the content team preferred to create a new Subject-specific to the layout they wanted than to re-use a subject.
  - Only 27 of the 450 subject entries are used more than once.
- Having the content spread between two entries is harder to see the big picture than to just have the fields embedded into the main entry.
- Some Modules only care about part of a Subject so it would be less confusing to give them their own entry type enforcing the fields it cares about.

All fields have help text explaining what the field is for and how it should be used.

We created an internal doc showing each Module type and how the fields map to the visual and presented it to the content team.

# What we learned

Keep things as simple as possible

Keep open lines of communication early and open

34



## LESSONS LEARNED

Don't over-engineer or try to future-proof

- We should have put less emphasis on re-usability of the 'Subjects' and more emphasis on making the content types simple and easy for the end-user to use.
- Better to have specific validations on the fields and enforce required fields than to have it ambiguous and dependent upon the context it is used.

There is a balance between allowing things to be reusable and intuitive

Open lines of communication early and often.

- Probably won't get the content model right the first time.
- Don't sweat when you need to make a change - Migrations scripts

**#1** Prioritize the needs of the primary user; the **content creator**.

**#2** Find the **balance** between reusable content types and overly specific content models.

## Audience Q&A



Rachel Church  
Sr Software Engineer



Amelia Collins Winger-Bearskin  
Developer Evangelist

36



A: Please type your questions in the Q&A box

1. How have you used migration scripts?
2. What's on the horizon (what's next for Clover) [We hear you're checking out space pace environments?]
3. Help center is going from static to dynamic - do you have any tips for listeners considering doing the same?

<https://docs.google.com/document/d/1KswixnH3EKYkCvxOMgOPyIDCwnQg-xT4dIXkIpYN08k/edit?usp=sharing>

Backup Questions:

- If you could re-design the content model for help app again what would you do differently?
-

# Thank you!

Keep an eye on your inbox for the slides and video

