



Customized Storefronts in DPS

Isaac Strack - Design Technologist



What is a Storefront?

Definition:

A Storefront is a flexible, efficient platform for marketing, distribution and retail sales.

Storefronts are used to:

- Control Distribution (Subscriptions, Paywalls, Authentication)
- Publish Content in App form quickly, on your own terms
- Promote Media Consumption and Consumer Loyalty
- Filter Results / Search
- Leverage Existing Future Content and Technology



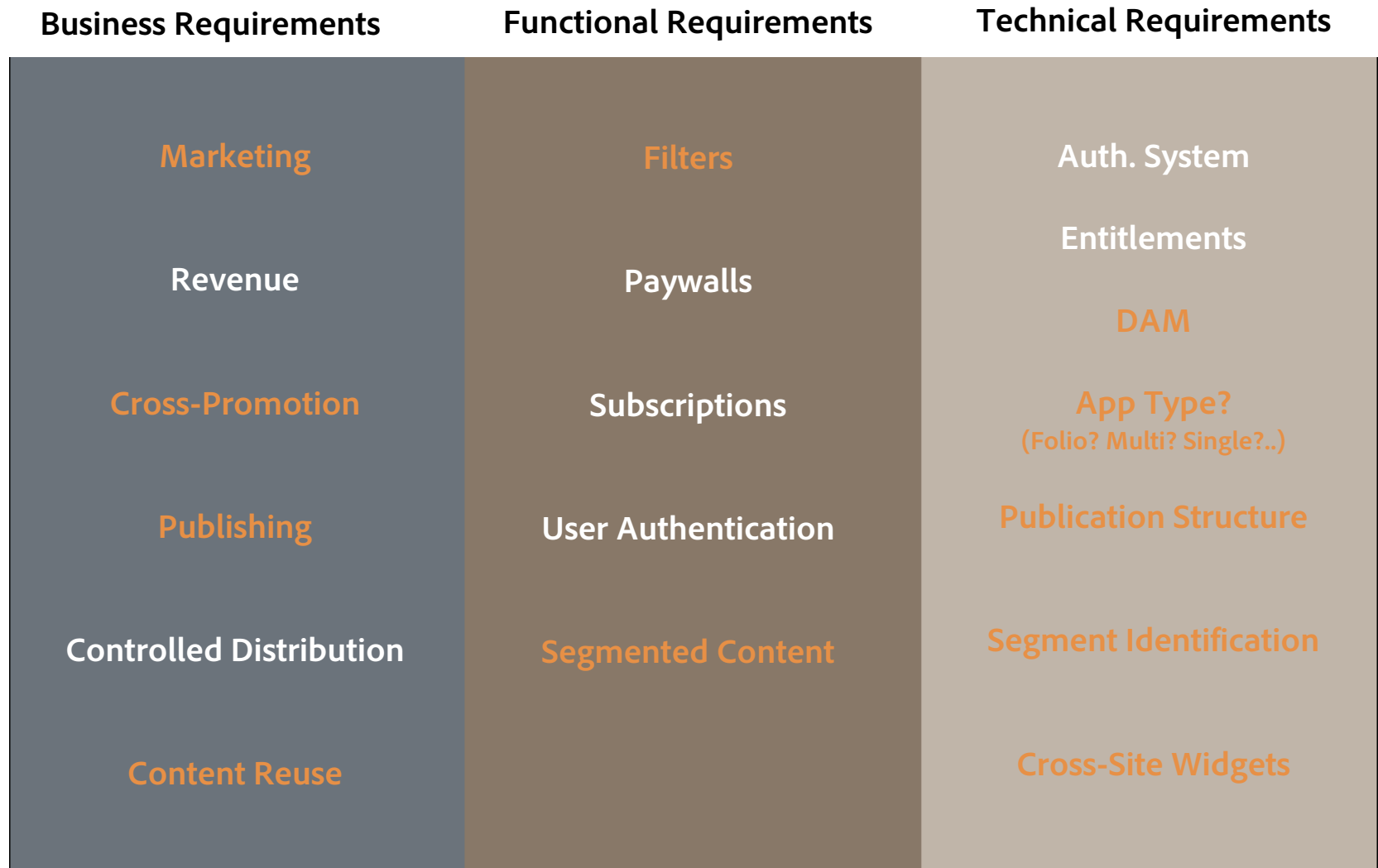
MARTHA STEWART LIVING

Sotheby's

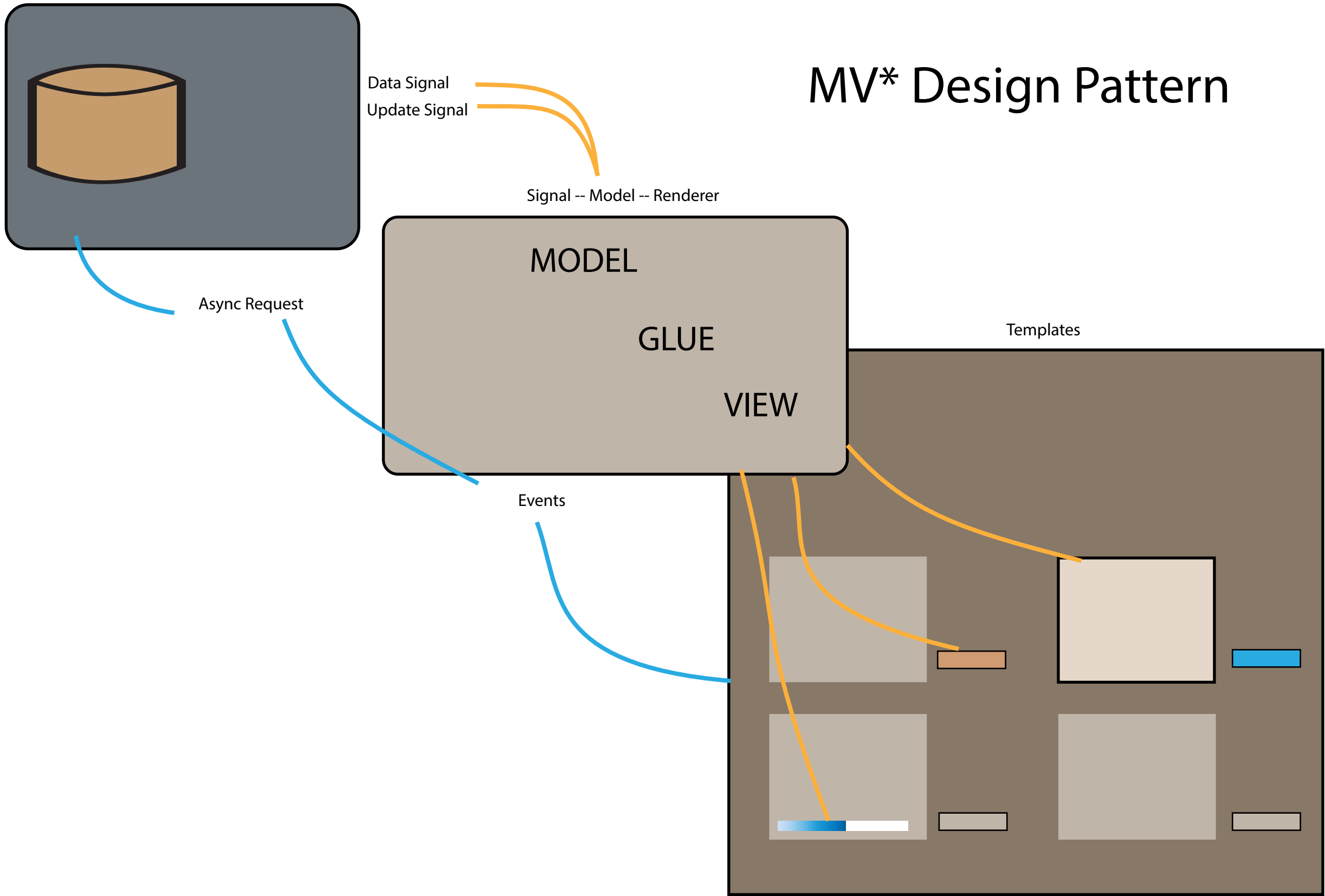
Storefront Development Process

Business Requirements	Functional Requirements	Technical Requirements
Marketing	Filters	Auth. System
Revenue	Paywalls	Entitlements
Cross-Promotion	Subscriptions	DAM
Publishing	User Authentication	App Type? (Folio? Multi? Single?..)
Controlled Distribution	Segmented Content	Publication Structure
Content Reuse		Segment Identification
		Cross-Site Widgets

Storefront Development Process



MV* Design Pattern



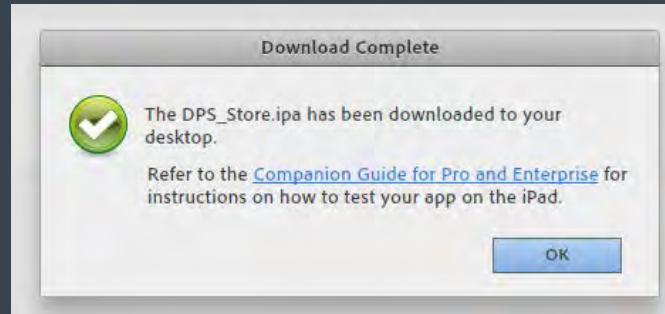
Simultaneous Development Tracks

Design Visualizations	Develop Structure
Basic Layout	1 - Dial Tone
	2- Set online/ offline Modes
Responsive Elements	3- Core Tech Requirements
Refined CSS	- Visualizations

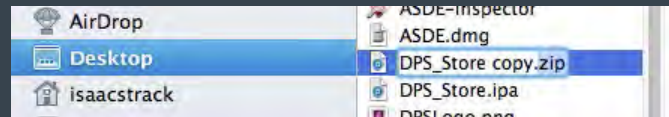
1- Dial Tone

Start with a Simple Template, and use a Shim

Create your .ipa



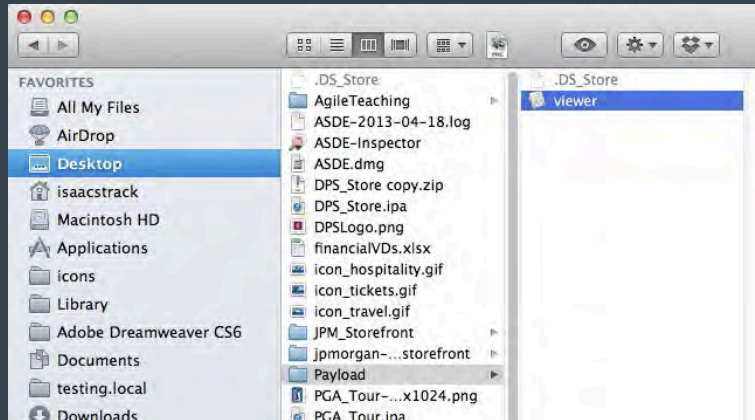
Make a Copy, and change to a .zip



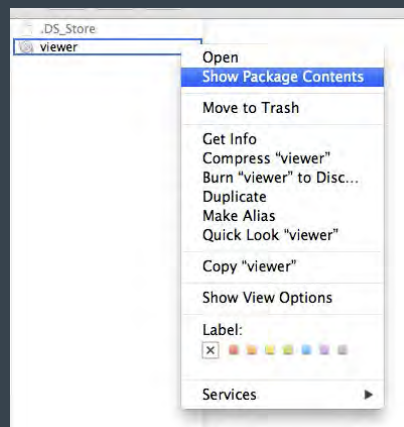
1- Dial Tone

Start with a Simple Template, and use a Shim

Unzip, Locate Viewer



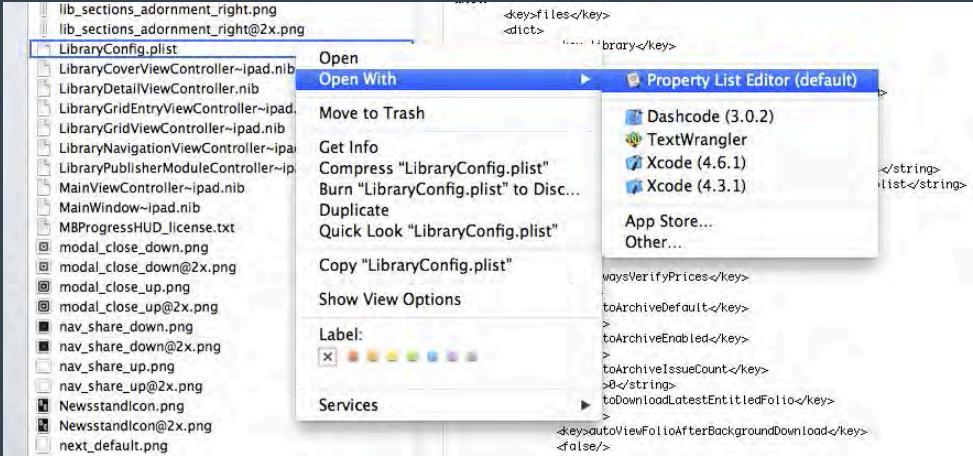
Show Package Contents



1- Dial Tone

Start with a Simple Template, and use a Shim

Locate LibraryConfig.plist




Copy service ptions fulfillment login ptions accountId:

Key	Type	Value
Root	Dictionary	(6 items)
files	Dictionary	(1 item)
generalSettings	Dictionary	(27 items)
rateThisAppOptions	Dictionary	(4 items)
serviceOptions	Dictionary	(3 items)
configuration	Dictionary	(2 items)
fulfillment	Dictionary	(4 items)
loginOptions	Dictionary	(2 items)
accountId	String	78e854ef3b4342d8a49f3fc51a898fb0
type	String	account
name	String	adobe_hosted_v1
serviceAuthUrl	String	https://origin.adobe-dcfs.com/ddp/issueServer/
serviceUrl	String	http://edge.adobe-dcfs.com/ddp/issueServer/
internalEntitlement	Dictionary	(5 items)
socialServiceOptions	Dictionary	(1 item)
socialSharingOptions	Dictionary	(6 items)

1- Dial Tone

Start with a Simple Template, and use a Shim

Test Account ID in a Browser:

 lighthouse.adobe.com/dps/v2_library_store_templates/fulfillment_proxy.php?accountId=78e854ef3b4342d8a49f3fc51a898fb0

Verify Valid ML:

```
<?xml version="1.0" encoding="UTF-8"?>
<results status="SUCCESS" message="Success">
  <issues>
    <issue id="370e3d70-d646-4787-aabe-95d56c4da09a" productId="com.adobe.dmx.customized.storefront.demo" formatVersion="2.0.0" targetViewer="25.0.0"
    hasSections="false" version="1" subpath="">
      <filter>Creative Suite</filter>
      <magazineTitle>Storefront Demos</magazineTitle>
      <issueNumber>6</issueNumber>
      <publicationDate>2013-04-22T06:00:00Z</publicationDate>
      <description>Adobe Creative Suite DPS Storefront Demo</description>
      <manifestXRef>Creative Suite</manifestXRef>
      <state>production</state>
      <libraryPreviewUrl landscapeVersion="1" portraitVersion="1">
        http://edge.adobe-dcfs.com/ddp/issueServer/issues/370e3d70-d646-4787-aabe-95d56c4da09a/libraryPreview
      </libraryPreviewUrl>
      <brokers>
        <broker>noChargeStore</broker>
      </brokers>
    </issue>
    <issue id="f631a8d4-f197-4a8b-840a-01cfe17cec53" productId="com.adobe.dmx.customized.storefront.demo" formatVersion="2.0.0" targetViewer="25.0.0"
    hasSections="false" version="1" subpath="">
      <filter>Creative Cloud</filter>
      <magazineTitle>Storefront Demos</magazineTitle>
      <issueNumber>3</issueNumber>
      <publicationDate>2013-04-22T06:00:00Z</publicationDate>
      <description>
        Adobe Web Experience Management DPS Storefront Demo
      </description>
    </issue>
  </issues>
</results>
```

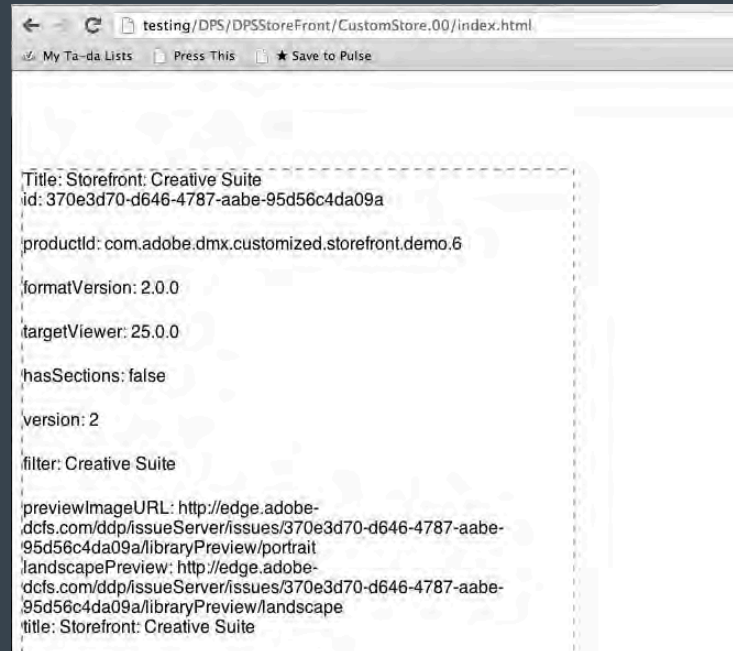
1- Dial Tone

Start with a Simple Template, and use a Shim

Add URL to Shim in main.js:

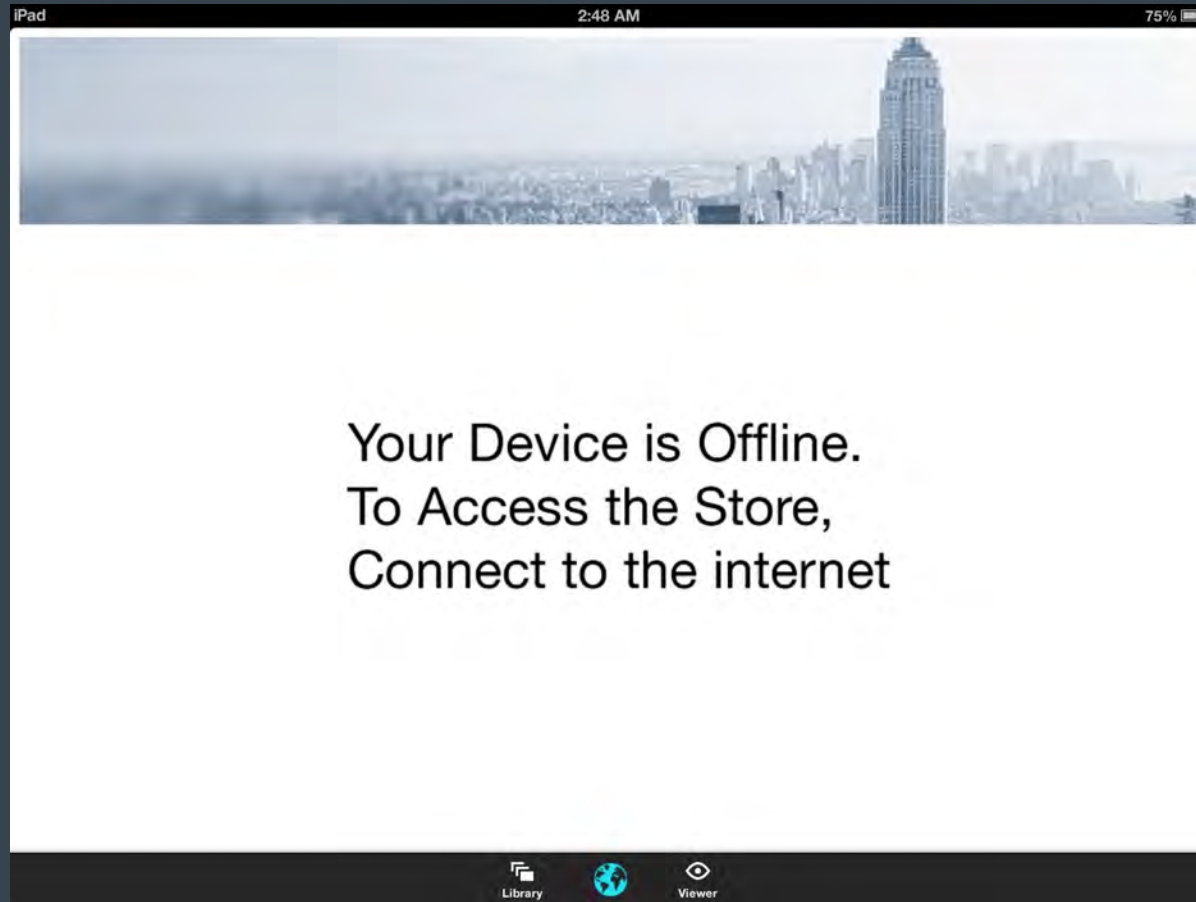
```
// Check to see if there is an internet connection.
$.ajax({
  type: "POST",
  datatype: "xml",
  // NOTE: the accountId is critical to getting the correct folio library.
  // You will need to modify this accordingly for your new application.
  url: "http://lighthouse.adobe.com/dps/v2_library_store_templates/fulfillment_proxy.php?accountId=78e854ef3b4342d8a49f3fc51a898fb0",
  data: "",
  success: function(xhresp,stat) {
    // Set a flag for the API availability in the ADOBE namespace.
    ADOBE.isAPIAvailable = isAPIAvailable;
  }
});
```

Test Shim in a Browser:



2- nline / fffline Modes

Use an intentionally bad URL, or turn off internet to your device:



3- Core Tech Requirements

Add AdobeLibraryAPI.js

```
21
22
23 !-This is the Adobe storefront library API code. Comment it out for dev testing -!
24 <script type="text/javascript" src="AdobeLibraryAPI.js"></script>
25
26 ! These are the folio-specific js files, used to create the structure on the page. -!
```

Implement Signal Listeners

```
if (!isAPIAvailable) return;
// Add a listener for when folios are added. This does not correspond to when
// a new folio is pushed rather when the viewer is aware of new folios.
adobeDPS.libraryService.folioMap.addedSignal.add(function(folios) {
    for (var i = 0; i < folios.length; i++) {
        ADOBE.liveController.addFolio(folios[i]);
    }
});
```

Use Remote files

```
!-These are the folio-specific js files, used to create the structure on the page. -!
!-Remote files are listed here: -!
<script type="text/javascript" src="http://too11.com/dps/shim.js"></script>
<script type="text/javascript" src="http://too11.com/dps/renderer.js"></script>
<script type="text/javascript" src="http://too11.com/dps/controller.js"></script>
<script type="text/javascript" src="http://too11.com/dps/startuphelpers.js"></script>
<script type="text/javascript" src="http://too11.com/dps/main.js"></script>
```

Compile and Redeploy .ipa

Add Features one at a time.


- Filter (segment)
- Banners (DAM)
- Download/ pen
- HTML idgets (if desired)

- Visualizations


Create your CSS. Test. Repeat.

[Creative Suite](#) [Creative Cloud](#) [Analytics](#)

Adobe DPS




<http://bit.ly/DPS-Store>



Storefront: Creative Suite

Adobe Creative Suite DPS Storefront Demo


Code: production




Storefront: WEM

Adobe Web Experience Management DPS Storefront Demo

Code: production



Storefront: Adobe Edge



Storefront: DPS

Tips / Reminders

- Start with Development Plan (Business, Functional, Technical)
- Add Visuals LAST (get core functionality down first)
- Develop with Remote, Complete with Local.
- Use your Shim, but don't forget to turn AdobeLibraryAPI.js on.
- Keep clean separation between Model, View, Glue
- Use Browser for foundation, device for aesthetics / performance
 - debug console is your friend
- displog() for uncatchable errors in browser
- Don't forget custom Library, or Library Script
- Coordinate HTML widgets with web development.

Summary

Customized Storefronts are a huge advantage:

- Controlled Distribution
- Your own "App Store"
- Filter Results / Segmentation
- Sharable Content = Consistent Brand, Faster Time To Market
- Build Excitement / Cross Promote

The advantages gained from having a customized storefront make the initial time investment worth the effort.



Adobe

<http://bit.ly/DPS-Store>