

# Managed Content

Delivered to Your Native Apps

Anthony Rumsey ( [@planetrumsey](#) )

Computer Scientist, Adobe Systems



Hi there!

[arumsey@adobe.com](mailto:arumsey@adobe.com)

[@planetrumsey](#)

A photograph of a playground with two young girls running on the grass. The word "Compromise" is overlaid in large white text. The playground features a wooden structure with a red roof, a red toy car, and various slides and climbing equipment. The background shows a building with a corrugated metal roof.

# Compromise

# AEM Apps



managed content

+

proven open source tech

# Mobile Landscape



1 million+



1 million+



2700+



# Hybrid Sweet Spot

*So in classic disruption theory, the hybrid model is eating its way up through the complexity chain. I'd argue that the majority of information-based apps today can be successfully implemented through this approach with varying levels of the native/HTML split.*

David Heinemeier Hansson - Founder & CTO at Basecamp



# Compromise

Native	Web
<ul style="list-style-type: none"><li>• navigation</li><li>• <i>some</i> views</li></ul>	<ul style="list-style-type: none"><li>• information views</li><li>• updates</li></ul>



Adobe

# Demo



Adobe



# //TODO:

1

Create AEM  
Content

A circular icon with a green border and a white background. Inside, there are several overlapping circles of varying shades of green and white, creating a network-like pattern.

2

Augment  
Android  
App

A green Android robot icon with a white outline. The text "Augment Android App" is overlaid on the robot's body.

3

Integrate  
native &  
AEM

A white icon representing a code editor or document. It shows a stack of papers with lines of code and some text. The text "Integrate native & AEM" is overlaid on the papers.

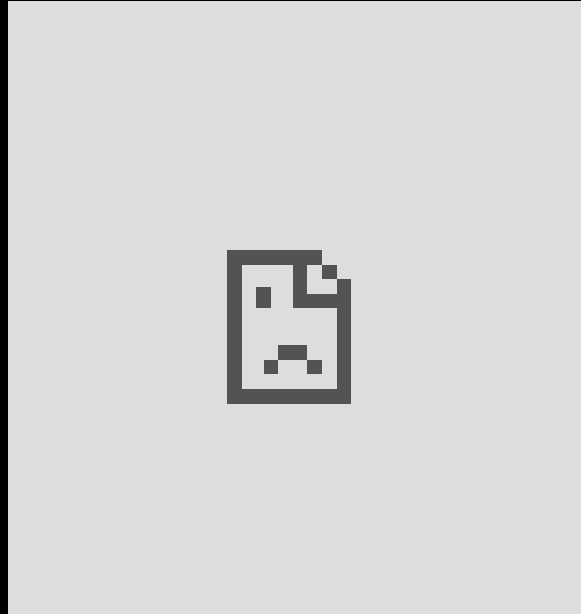
4

Plugins!

A white icon representing a printer or a similar device. It has a paper tray on the left and a paper output tray on the right. The text "Plugins!" is overlaid on the device.

# Create Content

- Design
- Components
  - splash page
  - home page
  - custom
- Content Sync
- Content



# Augment Android App

## Layout

```
<org.apache.cordova.CordovaWebView  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent"  
    android:id="@+id/aemWebView"/>
```

# Augment Android App

## Fragment

```
public class HomeFragment extends Fragment implements CordovaInterface {  
  
    CordovaWebView aemView;  
  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup container,  
        Bundle savedInstanceState) {  
  
        LayoutInflater localInflater = inflater.cloneInContext(new CordovaCon  
        View rootView = localInflater.inflate(R.layout.home_view_frag, contain  
        aemView = (CordovaWebView) rootView.findViewById(R.id.aemWebView);  
        Config.init(getActivity());  
        aemView.loadUrl(Config.getStartUrl());  
        return rootView;  
    }  
}
```

# Integrate

- Download content package
- Phonegap Build
- Copy Assets

# Plugins

```
public class Navigation extends CordovaPlugin {  
  
    @Override  
    public boolean execute(String action, CordovaArgs args,  
        final CallbackContext callbackContext) throws JSONException {  
  
        if ("navigate".equals(action)) {  
            this.cordova.getActivity().runOnUiThread(new Runnable() {  
                public void run() {  
                    navigateSync(args, callbackContext);  
                }  
            });  
            return true;  
        }  
        return false;  
    }  
}
```

{Code Time;}



Q & A

[bit.ly/aem-apps-native](http://bit.ly/aem-apps-native)

# References

- [Android + Phonegap sample with AEM](#)
- [Embedding a Cordova WebView into an Android fragment](#)
- [Mixing Cordova/PhoneGap Components with Native iOS](#)
- [Who's Winning, iOS or Android? All the Numbers, All in One Place](#)
- [Hybrid Sweet Spot](#)
- [PhoneGap](#)
- [PhoneGap Enterprise](#)