



## Create a Apple Universal Search Catalog feed POC

## Details

Type:	User Story	Status:	<span>DONE</span> (View Workflow)
Priority:	As Available	Resolution:	Done
Affects Version/s:	None	Fix Version/s:	None
Component/s:	Tigers		
Labels:	<span>Andrew</span> <span>Juampy</span>		
Epic Link:	<span>Universal Search</span>		
Sprint:	Sprint 40		
Story Points:	8		

## Description

## USER STORY

As a brand I need a way to provide my media catalog to Apple's Universal Search platform, so that end users can use tvOS, and other Apple products to search for video and play it where it's available.

## DESCRIPTION

As per Apple Universal Search requirements, we need to create a catalog feed (see attached documentation). We also want to be mindful that other Universal Search vendors will be required down the road, namely Amazon FireTV and Roku. As such we should architect a solution that would require the minimal amount of work to adapt to each platform.

For the initial POC, use draco\_udi Content titles as a data source for the media catalog. The solution should be architected in such a way that the catalog feed uses UDI Content title data as a basis, but allows brands the opportunity to modify/augment the data set to come to a complete set of data required for all Universal Search vendors. The solutions end product is the generation of a valid catalog feed XML document for Apple Universal Search.

See [DRACO-2441](#) for more details on the discovery of the Universal Search feature.

## RESOURCES

- [http://docs.turner.com/display/SASDCMS/Draco+Requirements?preview=/86358009/94245720/tvOS\\_Map\\_080316-03.docx](http://docs.turner.com/display/SASDCMS/Draco+Requirements?preview=/86358009/94245720/tvOS_Map_080316-03.docx)

## TestRail: Results

You are not yet logged in to TestRail. Please log in to use the integration.

[Log in to TestRail](#)

## Attachments

apple-universal-search.xml	202 kB	10/May/17 4:35 PM
>  Universal Search Feed Work.zip	776 kB	02/May/17 1:06 PM
Universal Search - Partner Onboarding Spec TBS.pdf	165 kB	02/May/17 1:06 PM

## Issue Links

is related to

DRACO-2507 Implement a way to exclude incomplete items from Universal Search Feed	<span>NEW</span>
DRACO-2508 Create a Roku Universal Search Feed	<span>NEW</span>
DRACO-2509 Create an Amazon FireTV Universal Search Feed	<span>NEW</span>
DRACO-2510 Create a Admin Configuration Page for Universal Search	<span>NEW</span>
DRACO-2511 Implement a way to alter the query for content title entities used for Universal Search	<span>NEW</span>

Show 4 more links (2 is related to, 2 relates)

## Activity

✓ Oliveira, Matthew [X] (Inactive) added a comment - 08/May/17 11:00 PM

Happy with my progress so far, I've put together the start of a POC module:

[https://github.com/turnercode/draco\\_universal\\_search](https://github.com/turnercode/draco_universal_search)

Have it to a point where it is spitting out a partial XML feed of Catalog items. Some highlights:

- Implemented REST endpoint for Apple Universal Search Catalog feed
- Implemented normalizers for Content title entities that source the XML feed.
- Feed is using values from Content title entities from Draco UDI

Still to do:

- Config page for global Apple Universal Search values (teamId, title, description, catalogId).
- Method for other modules to alter the XML
- At least start a gap analysis between the data given by Content titles and the required data for Apple Universal Search
- Think about how other Universal Search feeds will plugin with minimal extra work

✓ Oliveira, Matthew [X] (Inactive) added a comment - 10/May/17 10:52 AM

As an update, I'm working on coming up with a model for the Universal Search data that can be used as an adaptor to generate the various XML feeds required by Apple, Amazon and Roku, which implement very different schemas.

In so doing I realized that we'll be missing a couple key data points to make the feeds validate:

- Offer data. This is how the media is available (Free, Subscription, Rental, Paid) as well as pricing.
- Locator data. This is where the media is available, typically deep links to play the content in the appropriate apps.

I'll proceed implementing the POC without these, the idea being brands will be able to plugin to the module to supplement, but this should be kept in mind. Although this will be the minimum required from brands to get valid feeds, there are other optional fields that will probably be desired, namely artwork.

I'm gathering all this data in a spreadsheet:

[https://docs.google.com/spreadsheets/d/1E9B4gJ2VxeEp4fNiaUFkvKkCSorV9PiRdrBuLD\\_YJNQ/edit#gid=0](https://docs.google.com/spreadsheets/d/1E9B4gJ2VxeEp4fNiaUFkvKkCSorV9PiRdrBuLD_YJNQ/edit#gid=0)

This is kinda gnarly to make sense of, unless your in my head, but basically this is a map of fields in each services schema that are available, mapping the equivalents in each row. I'm tracking where we can source each field. Required fields are in red. There is a "Required from Brands" column to call out fields that are either required or suggested. This is incomplete, I'm working to fill out details for TV shows, seasons, and episodes as well.

✓ Oliveira, Matthew [X] (Inactive) added a comment - 10/May/17 4:35 PM - **edited**

Trucking along.

Today I finished designing and implementing an object model I like for Universal Search feed and corresponding feed items that should fit the requirements for all three provider schemas. I also used it to generate a partial Apple Catalog feed. So far I've covered Movies. Here is an example of the generated XML I'm getting so far:

[apple-universal-search.xml](#)

✓ Oliveira, Matthew [X] (Inactive) added a comment - 12/May/17 4:14 PM

OK I've written a lot of code, and I'm satisfied with the POC I've got now:

[https://github.com/turnercode/draco\\_universal\\_search](https://github.com/turnercode/draco_universal_search)

Here's some bullet points:


- Depends on UDI data, specifically is based on Content Title entities.
- Declares a single REST resource to return the Universal Search Feed. Which format you want, is controlled by the `_format` param. I've declared custom formats as applicable, eg.

```
/universal-search-feed?_format=apple_us_catalog
/universal-search-feed?_format=roku_us
```

Eventually we'll end up with:


```
/universal-search-feed?_format=apple_us_catalog
/universal-search-feed?_format=apple_us_availability
/universal-search-feed?_format=roku_us
/universal-search-feed?_format=amazon_us
```

- Features an object model for Universal Search Feeds and Feed Items (media) that is a generic super set of all the fields needed across the three vendors.
- This generic object model is what is passed to brands to modify/augment. Thus brands don't need to care about the implementation details of each service provider, the module handles that completely.
- Currently outputs a partial Apple Catalog Feed, and Roku feed based on Content title entities imported from UDI
- Demo module available to show how brands can alter the output of the feeds.

✓  Oliveira, Matthew [X] (Inactive) added a comment - 12/May/17 4:32 PM

Novillo, Juan Here is a PR to facilitate the review: [https://github.com/turnercode/draco\\_universal\\_search/pull/1](https://github.com/turnercode/draco_universal_search/pull/1)


However, this is a brand new repository and I've been working on master. The branches involved above are just for the purpose of review. There isn't any need to finish with a merge.

✓  Oliveira, Matthew [X] (Inactive) added a comment - 15/May/17 10:44 AM

I've created a set of tickets to follow up on the work I started here:


DRACO-2507  
DRACO-2508  
DRACO-2509  
DRACO-2510  
DRACO-2511  
DRACO-2512

cc Bose, Santanu Firfer, Brandon

✓  Berry, Andrew added a comment - 15/May/17 2:06 PM

I've reviewed the PR above and left some notes. Looking very nice!

A /ton/ of this seems to be data definitions and such. It looks like it would be pretty straightforward to have a "universal\_search" module on d.o, with the draco module just being the UDI specific implementations.

✓  Oliveira, Matthew [X] (Inactive) added a comment - 16/May/17 8:26 AM

Thanks for the review [Berry, Andrew](#)! Yeah, that's a good point, we could refactor it to have a common upstream piece on d.o. We can have that conversation for sure. We can handle that in future tickets. For now, I'm going to call this POC done.

## People

Assignee:



Oliveira, Matthew [X] (Inactive)

[Assign to me](#)

Reporter:



Firfer, Brandon

Watchers:

3 [Start watching this issue](#)

## Dates

Created:

02/May/17 1:04 PM

Updated:

16/May/17 8:27 AM

Resolved:

16/May/17 8:27 AM

## Development

[Create branch](#)

## Agile

Completed Sprint:  
[Sprint 40 ended 16/May/17](#)  
[View on Board](#)

TestRail: Cases

---

*You are not yet logged in to TestRail. Please log in to use the integration.*

Log in to TestRail

---