Episode Book

EPISODE #5862

One Show, Eighteen Feeds: Apple's Duplicate Content Rules

PUBLISHED	11 October 2026 at 11:05 UTC
DURATION	21:06
LISTEN	myweirdprompts.com/episode/podcast-topic-feeds-syndication-31219/
HOSTS	Corn, Herman
SCRIPT MODEL	deepseek-flash
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.3
SOURCES	13 cited by the research (listed at the end)
BOOK GENERATED	19 October 2026 at 17:40 UTC

SYNOPSIS

Daniel wants to submit eighteen topic-channel RSS feeds of a single podcast to Apple and Spotify. The problem: Apple's data model has shows and episodes, nothing in between — so eighteen feeds isn't eighteen views of one show, it's eighteen shows. This episode walks through Apple's duplicate content guidelines (1.9, 1.4, 4.2), the RSS requirements that make duplicate enclosure URLs silently ignored, and the fork where both paths — shared GUIDs or fresh ones — lead to failure. We also look at why the industry standard runs the opposite direction (network aggregation feeds like Captivate's Network Feed and Pantheon's sampler model), why multi-feed submission isn't framed as spam, and how auto-discovery can quietly index feeds you never submitted.

TRANSCRIPT



Corn

The show has quietly turned into a publishing house, and nobody agreed to that.



Herman

Eighteen channels. One master feed. Each channel with its own RSS feed, its own page on the site, its own little corner of the app.



Corn

And Daniel wants to know whether he can take all eighteen of those feeds and walk them up to Apple and Spotify like a man carrying eighteen pies into a restaurant.



Herman

That's the shape of it.



Corn

He built the channels so listeners could subscribe to just the technology episodes, or just the DIY ones, instead of drinking from the whole firehose. Sensible. Then he looks at the syndication side and starts wondering what happens when the same episode shows up in a topic feed and the master feed at the same time. He's worried about getting flagged as a spammer, about the submission grind being entirely manual, whether the duplication breaks compliance or search, and which syndicators would actually be suited to a multi-stream model.



Herman

Good prompt.



Corn

It's a good prompt. And the short answer is that the industry has a well-documented pattern for exactly this problem, and it is the precise inverse of what he's built.



Herman

Which is the whole episode, really. Everything else follows from that.



Corn

Then let's get into it. Herman.



Herman

The thing to understand first is that this was never a hosting question. Daniel thinks he's asking where to plug the feeds in. He isn't. He's asking what a show *is*.



Corn

In the eyes of a directory.



Herman

Apple and Spotify don't have a model for "one show with eighteen views." Their data model has shows, and it has episodes. That's it. There's no third entity where a topic channel would live. So eighteen topic feeds arriving at Apple isn't eighteen views of one thing. It's eighteen things.



Corn

Eighteen shows, each of which happens to contain episodes that are also in a nineteenth show.



Herman

Right. And the episode volume is what makes this urgent rather than theoretical. The catalogue is in the thousands. There's a new episode every day. The channels are assigned automatically at publish time, which is the correct engineering call, because no human team could classify at that rate.



Corn

And that's the irony. The publishing side got automated because the volume demanded it, and now the distribution side is the bottleneck, and distribution is the one part of podcasting that never automated anything.



Herman

The one part where a man still fills in a form.



Corn

So frame the tension for me, because I want it stated clearly before we start walking through documents.



Herman

The tension is that the same episode audio appearing in two feeds is either a feature or a violation, and it depends entirely on which document you read. From the subscriber's side, it's the point. Subscribe to technology, get technology. That's a better product. From Apple's side, it's two copies of the same episode, and they have a rule with its own number about that.



Corn

Which is where this stops being a product question and becomes a compliance one.



Herman

And the newspaper-family analogy Daniel reaches for is real. Networks with ten or twenty shows are completely normal, well-supported, and there's infrastructure built for them. But the analogy breaks in one specific place, and that's where the whole problem lives.



Corn

Then let's go there. The pattern everyone else uses.



Herman

The established pattern is aggregation, not slicing. Podcast networks solve discovery by taking many shows and pulling them into one feed. Captivate actually ships this as a product. It's called a Network Feed, and the pitch is that you create a network-level feed, submit the network to podcast apps, share it with advertising partners, and any podcast on your network automatically publishes its new episodes into that feed.



Corn

So one feed, many shows, automatic propagation.



Herman

And it's the exact opposite direction from what Daniel's doing. He has one show and eighteen feeds. The industry standard is eighteen shows and one feed.



Corn

The Pantheon case is the closest documented analogue, isn't it?



Herman

Closest and still opposite. Pantheon runs over a hundred music shows, and their host Peter Ferioli describes their strategy as a sampler feed, sometimes a magazine feed. His framing is that it's "the vertical look at categorization and serving niche verticals and trying to be everything to everyone."



Corn

Serving niche verticals.



Herman

Which is exactly Daniel's instinct. Pantheon wanted listeners who care about one niche to be able to find the shows in that niche. They just solved it by packing the niche together, not by unpacking the whole into eighteen pieces.



Corn

RSS.com's network guide says the same thing in plainer language. Running a network means juggling multiple feeds, each targeting different audience segments, and you manage them from one dashboard. And they recommend covering Apple, Spotify and YouTube Podcasts as the floor.



Herman

With the key word being "segments." Each feed in that model has its own content. It's not a filtered view of some other feed.



Corn

Lower Street puts it about as cleanly as it can be put. Each show in a network lives in its own feed but benefits from shared production resources and cross-promotion.



Herman

So the multi-feed family is completely normal. The thing that isn't normal is that all those feeds would be carrying the same episodes.



Corn

That's the break in the analogy.



Herman

That's the break. A newspaper with twenty podcasts has twenty podcasts. Twenty different audio products. Daniel has one audio product wearing eighteen hats, and the directories are going to ask to see the face.



Corn

So now the rulebook. Apple's own.



Herman

Apple has a guideline with a title that reads like it was written about this exact plan. It's 1.9, and it's called "Duplicate Content and Repeated Submissions." The text is short. "Content may be removed if multiple copies of a show or episode are submitted. If creators seek to publish an updated version of a show or episode, they should upload a replacement to the original."



Corn

"Multiple copies of a show or episode."



Herman

That's the sentence that matters. And note what the remedy is in the second half. If you want to publish an updated version, you upload a replacement to the original. There is no branch in that rule for a legitimate alternate view of the same episode. The only sanctioned way to have two of the same thing is to have one.



Corn

Hm.



Herman

There's a second one too. 1.4, Impersonation. "Podcasts designed to mislead audiences by mimicking, copying, or duplicating other content or search terms are not permitted."



Corn

Duplicating other content.



Herman

And then 4.2, Duplicate Shows, which sits under the paid content rules. "A show must be associated with a channel and duplicate shows are not permitted."



Corn

So that's three separate rules, in three different sections, all of which touch this. That's not an accident of drafting. They've thought about duplicates in at least three frames.



Herman

And here's the part I find interesting, because the policy is the soft layer. Underneath it is a mechanical layer, and the mechanical layer is much more specific.



Corn

The feed requirements.



Herman

Apple's RSS requirements say every episode must have a unique enclosure tag, and then this. "Apple Podcasts will ignore duplicate enclosure URLs."



Corn

Ignore.



Herman

Ignore. Not reject. Not flag. Ignore. And the second requirement is that all episodes must carry a globally unique identifier, a GUID, which never changes. The technical update Apple publishes for hosting providers sharpens that: "Apple Podcasts will not process new episodes with duplicate GUIDs."



Corn

So walk the fork, because I think this is the crux and I don't want it glossed.



Herman

There are exactly two ways for a topic feed to carry an episode, and both of them are bad.



Corn

Number one.



Herman

The topic feed reuses the master feed's GUIDs and the master feed's enclosure URLs. That's the natural engineering choice. Same episode, same identifiers, just a different subscription view. And the result is that Apple sees a duplicate enclosure URL and ignores it, and if the GUID duplicates a processed one, it doesn't process the new episode at all. The feed is valid. It's well-formed. It parses. And nothing lands. It looks alive from the outside and it's doing nothing.



Corn

Silently dropped.



Herman

Silently. No error, no notification. The worst kind of failure, because you'd have no idea.



Corn

Or number two, the feed mints fresh GUIDs and fresh enclosure URLs for the same audio.



Herman

And now every technical requirement is satisfied and you have walked straight into 1.9. Because what you have produced, in Apple's terms, are multiple copies of the same episodes. Fresh identifiers don't make them different content. They make them detectable duplicates with correct paperwork.



Corn

So one path gets ignored, the other path gets removed.



Herman

That's the fork. And I want to be honest about this, because a listener might reasonably expect us to find the third way, and I don't think there is one. You can make the GUIDs unique or you can make them shared. Those are the options. And Apple's documentation closes both.



Corn

So this isn't a case of the rules being unclear and someone finding a clever workaround.



Herman

It's a case of the rules being clear and the workaround being the thing the rules were written for.



Corn

And here's what I keep coming back to. You went looking for anyone who's done this, and there isn't anyone.



Herman

There isn't. And I want to be precise about that, because the absence is the finding. Nothing in Apple's creator documentation, nothing in Spotify's support pages, nothing from Captivate or RSS.com or Daniel J. Lewis or Podscan describes a publisher successfully submitting many topic-sliced feeds of a single show.



Corn

The documented pattern is always the other direction.



Herman

Always. Many shows into one network feed. Many shows into one sampler feed. Every published case, every host feature, every guide, runs that way. The one-show-to-many-feeds arrangement, at the syndication layer, appears to be unpublished.



Corn

Which is a strange feeling, because Daniel usually sends us things where the answer exists and we have to find it.



Herman

This one doesn't exist yet. And I'd rather say that plainly than manufacture a precedent.



Corn

So let's turn to the question he actually asked first, which is the etiquette one. Am I going to get flagged as a spammer.



Herman

And the answer is no, not in the way he means. We searched both Apple's and Spotify's documentation specifically for the spam framing, and neither of them frames multi-feed submission as spam. The word doesn't appear in this context.



Corn

That's a genuine surprise.



Herman

It reframes the whole problem. The risk isn't volume. Nobody at Apple is counting how many feeds one publisher submits and getting suspicious. The risk is duplicate listings. That's a different problem with a different remedy, and it's a problem about what ends up in the catalogue, not about how it got there.



Corn

Which makes the etiquette about not creating duplicates, rather than about how many times you knock on the door.



Herman

And there's a documented failure mode here that's worth walking through, because the mechanism is not intuitive.



Corn

Daniel J. Lewis.



Herman

Daniel J. Lewis, on The Audacity to Podcast, laid out how duplicates actually happen, and it's two routes. Either a secondary feed gets auto-discovered and indexed by crawlers, or the podcast gets resubmitted. And the auto-discovery one is the one that should worry Daniel, because he found eight separate RSS feeds auto-discovered for a single one of his own podcasts.



Corn

Eight.



Herman

Eight, for one show. Google's crawlers found eight different feeds pointing at the same content and indexed them. And his rule coming out of that is blunt. "Do not resubmit your podcast if it's already listed."



Corn

So the risk isn't that you submit eighteen and get punished. It's that sixteen of them get quietly indexed as their own shows whether you submitted them or not.



Herman

Which turns the whole etiquette question around. The etiquette isn't about submission discipline. It's about controlling what's discoverable, and you have much less control over that than people assume.



Corn

Now the part that makes this expensive.



Herman

Since iOS 14.5, Apple Podcasts and Spotify use proxy-based subscriptions. And this changes everything about cleanup.



Corn

Explain what that means, because it sounds like plumbing.



Herman

It's plumbing with a bill attached. Before, a listener who subscribed in Apple Podcasts was effectively subscribing to your RSS feed. Apple was a directory pointing at your URL. After 14.5, the listener subscribes to Apple's catalogue entry for the show. Apple serves the audio. Apple holds the subscriber.



Corn

So the listener isn't connected to the feed at all.



Herman

Not directly. And the consequence is that if a topic feed gets indexed as its own show and you later need to remove that listing, deleting it doesn't just remove a duplicate. It loses its followers. They subscribed to that catalogue entry. It goes away, they go away.



Corn

That turns a distribution mistake from a reversible experiment into a one-way door.



Herman

Which is the single most important practical consequence in this whole episode, and it's why I'd push back on treating the eighteen feeds as a thing you try and then tidy up. You don't tidy this up. Lewis's fix for a duplicate listing involves 301 redirects, an announcement feed kept alive for a minimum of three months, and changes at the portal level to the source feed. And his own summary of it is "it's messy and you might still lose some numbers."



Corn

Three months of a feed nobody wants, kept alive to tell subscribers the thing they're subscribed to is going away.



Herman

Minimum.



Corn

And Spotify's own guidance on this is thin to the point of being funny.



Herman

It's one line. "Duplicate versions: If you see multiple versions of your show on a platform, contact us to get one of the versions removed."



Corn

Which tells you exactly how they categorise it.



Herman

It's an error state. It's not a supported configuration. When Spotify sees duplicates, they treat it the way a bank treats a duplicate charge. Something went wrong, contact support, we'll clear it. There's no version of that sentence that reads as "yes, publishers do this deliberately."



Corn

Now the SEO half, because I think this one is unsolved and I want to make sure we're not overselling it.



Herman

It's unsolved, and the reason is structural. Google's canonicalization documentation explains the mechanism. Canonicalization "helps Google show only one version of the otherwise duplicate content in its search results." That's the goal, and for web pages there's a tool. You put a canonical tag on the page and you tell Google which version is the real one.



Corn

And RSS feeds have no equivalent.



Herman

No equivalent. There's no canonical tag in RSS. There's no mechanism by which Daniel can formally declare to a search engine that the technology feed is a non-canonical view of the master feed. The declaration doesn't have a syntax. You can't say it.



Corn

Search engines will do whatever they do, and he has no vocabulary to influence it.



Herman

There is a partial workaround, and it's worth naming because it's the closest thing that exists. Feeddoc publishes SEO guidance for syndicated media, and it recommends including a canonical underscore URL in every feed item. And it specifically warns about inconsistent metadata across RSS and Atom and JSON feeds. Timestamps, IDs or author names changing between feeds, which breaks entity mapping.



Corn

That's an item-level convention.



Herman

It's a convention. Feeddoc's recommendation is a good practice, not a standard the directories honour. It might help an entity-mapping system decide that two items are the same thing. It is not a declaration Apple or Google has agreed to respect.



Corn

Which is a real distinction, and it's the sort of thing that gets lost when people say "just use canonical tags." You can't. You can use something that looks a bit like one and hope.



Herman

It's worth separating that from the GUID question, because those get conflated. The canonical URL convention is a soft signal. The GUID is a hard one.



Corn

Talk about the GUID as an operational risk, because you flagged it earlier and I want it on the table as its own thing rather than a footnote to compliance.



Herman

The GUID is the identity of an episode. It's how every app knows that this is the same episode it downloaded last week. Podcast Host Lab puts the danger simply: change a GUID and every app treats that episode as brand new. Brand new. A fresh download, a fresh unplayed state, a fresh entry in the queue.



Corn

If anybody ever regenerates identifiers per feed, that's not a formatting choice. It's resetting the listener's relationship with the entire back catalogue.



Herman

PRX's feed requirements make the directory side of it explicit. "Ensure that there are no duplicate URLs in your feed. Apple Podcasts will ignore previously used URLs."



Corn

There it is again. Ignore.



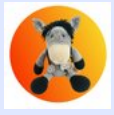
Herman

Ignore is the word Apple uses throughout. They don't argue with you. They just don't see it.



Corn

The second-order implication, and I'll state it and let you react to it. The proxy-subscription era means the cost of a distribution mistake is measured in lost subscribers, not just lost listings.



Herman

That's the correct reading.



Corn

Which argues for treating eighteen feeds as a deliberate, small, high-demand set rather than a full fan-out. And it argues for asking the question Daniel might not want asked, which is whether eighteen topic feeds actually serve listeners better than one well-organised master feed plus a handful of channels.



Herman

I'd want to be careful there, because the channels work. They exist, they classify automatically, they're live on the site and in the app, and a listener who wants only the technology episodes already has that today. The channels are not the problem. The problem is the decision to mirror all of them outward into syndication.



Corn

The internal construct is fine. It's the external submission that carries the cost.



Herman

That's a much narrower question, which is good news. He's not being asked to undo anything. He's being asked how many of the eighteen are worth the exposure.



Corn

Which leaves the last thing he asked, and it's the one where the honest answer is a bit deflating. Which syndicators are actually well suited to a multi-stream model.



Herman

Nobody in the documentation names a directory that explicitly supports one-show-many-feeds. So the plain answer is that the platforms suited to it are the ones that let you manage many feeds from one dashboard and automate distribution, rather than the ones that maintain curated catalogues.



Corn

Captivate and RSS.com being the examples there.



Herman

Captivate because the network feed construct is a real product feature, and RSS.com's guidance because it's built around managing multiple feeds from one dashboard. Those are the hosts whose tooling assumes you have a family of feeds. That's a hosting answer, not a syndication answer, but it's the closest thing to a real answer that exists.



Corn

The two most likely to treat the fan-out as duplicate content are Apple and Spotify.



Herman

The two biggest. Which is the awkward shape of this. The platforms with the most listeners are the ones with the most explicit rules against the arrangement, and the tooling that supports it sits one layer down, at the host.



Corn

Hold on. I want to go back to something, because I think we let it pass too quickly.



Herman

Go on.



Corn

You said Apple won't process new episodes with duplicate GUIDs. Won't process.



Herman

It's the stronger word, and I think it's deliberate.



Corn

The failure is upstream of the catalogue. The episode doesn't get rejected from a listing. It doesn't get as far as being considered.



Herman

That's why I keep saying silent. There's no rejection to appeal. There's nothing to point at and say "but this one is legitimate." The system looked at it and moved on.



Corn

That's worse than a block. A block at least tells you where the door is.

SOURCES

What the research for this episode read before the script was written. Primary sources first.

1. **Apple Podcasts for Creators [primary]**
   Content guidelines (rule 1.9 Duplicate Content, 1.4 Impersonation, 4.2 Duplicate Shows), fetched 2026-10-11
   https://podcasters.apple.com/support/891-content-and-subscription-guidelines

2. **Apple Podcasts for Creators [primary]**
   Podcast RSS feed requirements (duplicate enclosure URLs ignored, GUID required), fetched 2026-10-11
   https://podcasters.apple.com/support/823-podcast-requirements

3. **Apple Podcasts for Creators [primary]**
   Technical updates for hosting providers (duplicate GUIDs not processed)
   https://podcasters.apple.com/4115-technical-updates-for-hosting-providers

4. **Spotify for Creators [primary]**
   Distributing your show to other platforms (duplicate versions guidance), fetched 2026-10-11
   https://support.spotify.com/us/creators/article/distributing-your-show-to-other-platforms/

5. **Captivate Help [primary]**
   How to Create and Use a Network Feed (updated 2024-12-17)
   https://help.captivate.fm/en/article/how-to-create-and-use-a-network-feed-1rx3u7l/

6. **The Audacity to Podcast**
   How to Fix Duplicate Listings in Apple Podcasts (2024-01-31)
   https://theaudacitytopodcast.com/how-to-fix-duplicate-listings-in-apple-podcasts-or-other-podcast-apps/

7. **RSS.com**
   How to Start a Podcast Network (published 2026-08-09)
   https://rss.com/blog/how-to-start-a-podcast-network-a-step-by-step-guide/

8. **Podscan / Podcast Network Insights**
   Curating Discovery Through Sampler Feeds | Pantheon Media (published 2026-03-26)
   https://podscan.fm/podcasts/podcast-network-insights-understanding-the-business-of-podcast-networks-1/episodes/curating-discovery-through-sampler-feeds-pantheon-media

9. **Google Search Central**
   What is URL Canonicalization
   https://developers.google.com/search/docs/crawling-indexing/canonicalization

10. **Feeddoc**
    Entity-Based SEO for Syndicated Media (canonical_url in feed items)
    https://feeddoc.com/entity-based-seo-for-syndicated-media-how-to-keep-search-sig

11. **Podcast Host Lab**
    What Is a GUID in a Podcast Feed?
    https://podcasthostlab.com/what-is-a-guid/

12. **PRX Help Desk**
    Overview of feed requirements (no duplicate URLs)
    https://help.prx.org/hc/en-us/articles/360023748473-Overview-of-feed-requirements

13. **Lower Street**
    What Is a Podcast Network? [2026]
    https://lowerstreet.co/blog/podcast-networks