Episode Book

EPISODE #5865

Four Pocket Players That Won't Sell You a Device Today

PUBLISHED	11 October 2026 at 11:47 UTC
DURATION	34:14
LISTEN	myweirdprompts.com/episode/pocket-audio-players-2026-06945/
HOSTS	Corn, Herman, Hilbert
SCRIPT MODEL	deepseek-flash
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.3
SOURCES	14 cited by the research (listed at the end)
BOOK GENERATED	19 October 2026 at 17:40 UTC

SYNOPSIS

Daniel wants an iPod-shaped appliance that holds a hundred gigabytes of podcast archive, plays it over Bluetooth at the beach, and never sends him a notification. The hardware exists — Mono, Fonix One, Sudo, and DuRoBo Krono are all building some version of it — but the market is stuck between waitlists, deposits, and concept reveals. This episode works through the real fork in the build: an ESP32 microcontroller with no operating system at all, versus open Android with all the baggage that comes with it. The deciding evidence is a documented history of Android storage frameworks breaking podcast apps, and the numbers on what a hundred-gigabyte library actually costs in cards, hours, and watts.

TRANSCRIPT



Corn

There's a particular kind of hot shame in having four hundred open browser tabs of things you meant to read, and Daniel knows it. Years of prompts shot off at lunchtime, answered, saved to a notepad, never opened again. His original fix wasn't a device at all. It was this show. Record the question, let the pipeline turn it into audio, listen at the gym.



Herman

And the seed worked. But the seed left a hole in the middle of it, which is the player.



Corn

Right. Because the format that finally works for him is audio out of a pocket, and the only machine he owns that does it is a phone that also does everything else. So he needs the appliance. Wi-Fi, Bluetooth, no notifications, no cellular if he can help it, enough storage to hold this entire archive, and build specs for a V2, hardware layer and software layer both.



Herman

The V1 was his own idea, an SBC in something shaped like a radio, and he's walked away from it. Not because it wouldn't work. Because he wants to take this to a beach and he isn't willing to be the man with the boombox.



Corn

He was very clear about the boombox. I don't think he's ever forgiven that guy.



Herman

And the new brief is an iPod. That rectangle. That's the target, and here's the part that makes it a real episode instead of a wish list. He asked whether someone has made this, and the honest answer is that four different companies are making it right now and not one of them will sell you one today.



Corn

Which is a strange kind of correct. His instinct that somebody must have built it is right. His timing is wrong by about a year.



Herman

So the landscape. Mono is the closest thing to his brief on paper. Pocket-sized, vintage-radio on the outside, and on the inside a four point two inch E-Ink screen that doesn't glow, a tactile joystick, a magnetic volume wheel, Wi-Fi sync, Bluetooth, USB-C. Their line is that it syncs once, listens offline, then disconnects. No notifications, that's their own phrasing. And it's a waitlist. Kickstarter in the first quarter, shipping targeted for the third quarter, so today you can give them your email and nothing else.



Corn

The one that actually takes money is Fonix One. Ninety-nine dollars, two point eight inch touchscreen, up to two terabytes on a microSD, sixteen plus hours, Bluetooth. And the interesting choice, it's an ESP32. One chip, no operating system worth the name.



Herman

No accounts, no apps, no algorithms. It runs its own little web server, so you connect it to Wi-Fi, open any browser on any machine in the house, and manage the library from there. Subscribe to a feed, upload files, all through the browser. It decodes MP3, FLAC, AAC, M4A, WAV, Opus.



Corn

You subscribe from a browser. To a thing that has no browser.



Herman

That's the whole trick of it. Then Sudo is the nostalgia play, New Matter's device, and it's the one that actually looks like the iPod. Hundred and three by fifty-nine by eight and a half millimetres of aluminium, a two point seven inch screen at three twenty by three twenty, and a physical scroll wheel, which nobody has shipped in fifteen years. Sixty-four gigs internal, microSD to a terabyte, headphone jack, twenty plus hours. Two hundred and forty-nine dollars, fifty dollar refundable deposit. Shipping late Q4 of next year.



Corn

Deposit.



Herman

Deposit. And a reviewer who went hands-on with the reservation page noted there are no photos of a working unit, no team names, and no audio specifications, and called it closer to a concept reveal than a product launch.



Corn

Two hundred and forty-nine dollars for a concept with a scroll wheel. I've bought worse. I want to be clear that I've bought worse.



Herman

Then there's the odd one out, the DuRoBo Krono. E-Ink tablet, open Android thirteen, six gigs of RAM, hundred and twenty-eight gigs of storage, Wi-Fi, Bluetooth, stereo speakers, built-in player, two seventy-nine ninety-nine. It isn't shaped like an iPod and it isn't trying to be, but it's the only device on that list you could actually load Podcast Addict onto today.



Corn

Which means the market framing only gets you halfway. Because the Amazon search returned something, and what it returned wasn't this. Mighty plays music without a phone or a screen, and Mighty is music-only, and Spotify support on it ends in April of next year. StorybuttonGo says no Wi-Fi, no ads, just stories, and it ships with a hundred and fifty preloaded originals.



Herman

Preloaded content. That's an appliance for someone else's library. Same with the rest of that shelf. They're streaming endpoints or they're little libraries that came sealed, and neither one is going to pull this show's RSS feed onto a card in Daniel's pocket.



Corn

So the entire category proves his point by failing it. Here's what I want to get into, because I think there's a fork in this build that decides everything downstream. The chip.



Herman

Espressif on one side, Android on the other, and they're not points on a spectrum. They're two different philosophies of what a computer is. The ESP32 is a microcontroller. No Google, no Play services, no background processes deciding it's time to check something. You flash it once and it's a fixed appliance, and the entire attack surface in your pocket is a web server on your home network and a speaker.



Corn

And the good part of that isn't just the lack of notifications. It's that a device with no OS doesn't accrete. Nothing installs itself on it. Nothing updates itself into a worse version of itself at two in the morning.



Herman

But that's also the cost, and it's the cost Sudo is going to run into. A closed OS means no F-Droid, no Podcast Addict, no third-party anything. Whatever the manufacturer shipped is what you have forever. If the feed format holds, fine, you own a player for ten years. If it doesn't, you own a very nice rectangle.



Corn

And Android gives you the opposite trade. You get the ecosystem, and you get Google's storage model, and you get Play services. Which sounds like the thing Daniel explicitly said he doesn't want.



Herman

He said ideally not even Google, and if he's picking, F-Droid as the manager, and I think that instinct is right, but I want to put a number on it because the number is the argument. Podcast Addict's own changelog is the clearest record we have. In 2021, Google forced apps onto a framework, and the developer pulled custom-folder storage out of the app entirely because the framework was, in his words, between twenty and a hundred times slower than direct file access. And also incomplete.



Corn

Twenty to a hundred times.



Herman

Between the two. And that's the framework Google said to use for files. So the app went backwards. Then in 2022 it came back, and they restored the ability to store downloads in a public folder. So this fight has a history, and it ran in both directions.



Corn

Meaning the worst outcome already happened once, on this exact app, for this exact reason, and the developer is still complaining about it in public changelogs. That's not a hypothetical risk for a hundred gigabyte archive. That's the documented pattern.



Herman

Podcast Addict has adapted, to be fair. There's a new Storage usage view that shows you which shows are eating the most space, and there's a Transfer downloaded files feature that moves downloads between devices over Nearby devices, though that one needs Android thirteen or newer.



Corn

Define eating. In real terms. Because the average person hears a hundred gigabytes and thinks that's absurd, and I want to check it.



Herman

This show publishes daily. Call an episode thirty to forty megabytes as an MP3 at a normal bitrate, say thirty-five to be conservative, times six thousand episodes. That's two hundred and ten thousand megabytes. Round it to about seventy to a hundred gigs depending on bitrate, whether there's artwork embedded, and how big the tags are.



Corn

A hundred gigabytes. Which in twenty twenty-six is what, a mid-tier laptop's worth of nothing.



Herman

It's two memory cards from a decade ago. It's nothing. Fonix takes two terabytes, Sudo takes one, and either one of those swallows the entire show, every episode ever recorded, thousands of times over. The whole archive fits on a card you can lose in a coat pocket for forty dollars.



Corn

The number that surprised me is the download time. Daniel said the first pull might be lengthy. How lengthy?



Herman

Sixty to a hundred gigs over ordinary household Wi-Fi at, I don't know, fifty megabits real-world throughput because the router's in the other room. That's on the order of five or six hours if it's moving steadily. So it's a set-it-going-overnight job, not a disaster. And after that it's incremental, a day's episodes, a few minutes.



Corn

So the scary part of the storage story is one evening. And then you're done. That reframes the whole constraint. It isn't whether a device can hold it. It's whether the software lets it.



Herman

Which is why the FTP test matters. If the file system on the device will accept a hundred gigs of audio submitted as files by an app that isn't allowed to write where it likes, you have a player. If it won't, you have a phone-shaped thing that holds ten episodes and a lot of opinions.



Corn

Let's do Bluetooth. Every one of those devices has it, and it's half of Daniel's required list, and I want to say why it's half.



Herman

Because the beach. Headphones. He was explicit, and it's the thread that runs through the whole prompt. The gym, walking, a weekend with the phone face down on the counter, and the hypothetical holiday by the sea with a few beers and a view. Bluetooth is what makes the thing a personal player rather than a nuisance, and Wi-Fi is the other half, because the download has to come from somewhere.



Corn

And cellular is the one he's actively trying to remove. Say the case for it, because I agree with him and I want to hear it built.



Herman

The case is that cellular is what makes a device into a phone-shaped thing. Once there's a SIM in it, the network can reach it. That's the definition of the reach, and the reach is the thing he's trying to delete. Wi-Fi only means the device only pulls feeds when you're at home, which is also the moment you're not using it, which is a elegant schedule. It downloads while you sleep.



Corn

And when you're at the beach, it's inert. It has nothing to give you but the library.



Herman

It's the same reason a paper book doesn't have a Home screen. There's nothing for it to summon you with.



Corn

I'll take that. The other thing, and I want to check this because it changes the build advice, is battery. Sudo claims twenty plus, Fonix claims sixteen plus.



Herman

Those are broadly the same class of number, and both depend on what the device is doing. Bluetooth is the drain. Sixteen hours with a modest codec and mid volume is a realistic long weekend, and the ESP32 has a real advantage there, because a microcontroller idling is drawing almost nothing.



Corn

Which matters if he's actually going to use it the way he says, which is this very specific afternoon where he's moving boxes and wants his brain occupied for hours. That's a form-factor decision as much as a battery one, though. And I want to put the form factor back on the table, because the two ends of this are different.



Herman

Mono is the anti-screen argument. A four point two inch E-Ink panel that doesn't glow, a joystick, a wheel you twist with a magnet in it. The screen is only on when you look at it, and it's black and white and it's not capable of being pretty. Nothing on it is designed to hold you.



Corn

And the two point seven three on Sudo is the small-screen argument. Pixels are cheap; restraint is architecture. A three twenty by three twenty display can only show you a list of episodes and a couple of lines about what you're playing, and that's a feature if you're the kind of person who can't be trusted with a rectangle.



Herman

It's the difference between a device with a screen and a device that can't avoid having one. And I think the practical answer for Daniel is a bit different from both, which is worth saying because he asked what he'd recommend, not whether he agrees.



Corn

Go on.



Herman

Buy the Fonix if he wants something now, and accept that it's marketed as a music player and that browser-based subscription is the whole management story. Put a card in it and it will hold the archive. Reserve the Sudo if the object itself is the point, and accept that you've put fifty dollars against a company that hasn't shown a working unit and is quoting the end of next year. Wait for Mono if he wants exactly the model he described, in which case he should be aware he's been waiting a while already.



Corn

And build nothing, given the memory market.



Herman

Right, and this is the part I'd put in the episode as the cost argument, because it's real. Seventy percent of high-end memory production is expected to go to AI data centers this year. Memory prices rose about fifty percent in the last quarter of twenty twenty-five, and TrendForce expects another seventy percent increase across this year. So everything that wants a card is getting more expensive.



Corn

Which makes a device you buy cheaper than a device you build, and it makes the Sudo's two forty-nine more uncertain than the deposit page makes it look. Because they priced it, and then the memory bill went up.



Herman

If they priced it in September of last year and shipped in late twenty-seven, the memory line in that build is a genuine question. Which is one more reason not to design your own board.



Corn

All right. Hardware's done, and I think we can say it's a solved problem with three partial answers. The software layer is where this gets hard, and it's also where Daniel has actually done the work.



Herman

He has, and I want to credit the shape of his plan, because he's right about the architecture. That's how you get a single-purpose appliance without writing thousands of lines of firmware. Borrow the feed parser. Borrow the download manager. Suppress everything else.



Corn

And the catcher he picks is one he already uses and trusts, Podcast Addict, which is the most feature-complete podcast app on Android.



Herman

It does nearly all of the download layer already. It can store shows on an SD card, there's a setting for the storage folder. It has Archive Mode, which is exactly the feature Daniel's use case needs, because Archive Mode downloads the older unplayed backlog, not just the newest episode that dropped, and it refills the space as you listen and as episodes get removed. And the new Storage usage view tells you which shows are heaviest.



Corn

And then the caveat, which I think is the most useful single fact in this whole episode.



Herman

There's a real one. With Archive Mode on, an unlimited Keep-at-most setting gets treated as a limit of two episodes.



Corn

Two.



Herman

Two, as a storage-protection measure. The app will not let you tell it to keep every episode ever, with Archive Mode enabled, and go about your day. It's an internal guardrail, it has a reason, and for a device whose whole purpose is holding the archive, it means the thing you actually want is something the app will not do out of the box.



Corn

So V2 has to either hold the archive as plain files outside the app's management, or it holds a working window of episodes, a few hundred, say, and refills from the network forever. Which is a different device from the one in the prompt.



Herman

And this is the collision with the storage restrictions, because the two paths react differently to Google's rules. If the app is managing downloads inside Android's storage framework, you're subject to the framework. If you're managing a hundred gigabytes of files yourself, as files, you're outside the app, and the app is only a player, and then you're not fighting anything, but you've also given up automatic refill.



Corn

Which is the actual design question. It's not Android or ESP32. It's managed by the catcher or managed by the user.



Herman

Daniel actually asked for the delete primitive, and that tells you where he is. A player with a delete button is a player where you decide what leaves.



Corn

My read on his read is that this is what he has been circling the whole time. He wants a fixed library he curates, in the same way he wants a fixed archive of six thousand episodes sitting on a card.



Herman

He wants the whole show in his pocket. That's the emotional core of the ask. Not a hundred latest episodes. The whole thing.



Corn

Then let's take the extensibility point seriously, because Daniel named it as the differentiator and I think he's right. His worry is specific. What if episodes get bigger, what if they get distributed some other way.



Herman

Fair worry. A hundred gigabytes was the right number this year. You could double it in a few years if bitrates rise, or if the feed starts carrying something richer. The device doesn't get to have an opinion about that.



Corn

The plan he proposes for it is to build a skeleton of an Android phone music player, restricted, and then add onto it. Which is the most boring possible answer and probably the correct one, because the boring answer is the one that still works when the format changes.



Herman

And the obvious host for it is the Krono. Open Android thirteen, six gigs of RAM, a hundred and twenty-eight gigs of storage, two eighty. It's not the shape he wants, it's about twice the device he needs, but everything he wants to do is installable on it today, and that matters when the alternative is a deposit and a date.



Corn

Let me put a fork in the whole discussion, though, because I think the memory spike is doing more work than we've given it credit for. If high-end memory costs are up fifty percent and heading up another seventy, then the smart play isn't a restricted device at all. It's to hold the archive somewhere boring and cheap and stream it to a dumb, capable player over Wi-Fi.



Herman

Hmm. A home server and a Fonix. You'd be running your own distribution.



Corn

Which is what he's actually doing already. It's a feed. The feed is the distribution.



Herman

That's a real point, and it survives the format change, because a feed can carry anything.



Corn

And then the pocket device never has to hold a hundred gigs, which means it never has to fight Android's storage model, which means the ESP32 route wins by not having the problem. I think that's the strongest version of this.



Herman

It's the strongest version of the software answer, and it's the one where F-Droid does the least work, which Daniel might not love.



Corn

He'll survive. He's an open source guy.



Herman

One more thing on F-Droid, because he names it as the app manager and I want to be accurate about what's in there. There are lovely minimalist offline players. Auxio, Gramophone, Fossify Music Player, Vibe, Lune, Tūī. They're small, offline, no accounts.



Corn

And every single one is a music player.



Herman

Every single one. They play files. None of them subscribe to an RSS feed, parse an XML document, and pull down new episodes on a schedule. So F-Droid can give you the playback layer and nothing else, and the catcher has to come from somewhere else, which means Podcast Addict, which means Play services aren't required but Google's storage rules are, because Podcast Addict is an Android app and Android is Android.



Corn

So the F-Droid plan gives you the front of the device and the app store, and the middle of the device, the bit that actually decides whether this works, is still Google's.



Herman

Unless you go the ESP32 route, in which case you never had a middle. The Fonix way is to make the browser the management and the firmware is a closed loop, and the day the feed format changes, you're hoping the vendor pushes a firmware update.



Corn

Which is not obviously worse than hoping the developer doesn't have to rewrite for a new storage API.



Herman

Daniel's instinct was to build something he could keep building on, and the truth is that the keep-building path is the Android path, and the Android path is the one with the wolf in it.



Corn

That's enough of a ledger to close on. But before we get there, I want to knock on the glass.



Herman

Whose glass?



Hilbert

Ninety-eight.



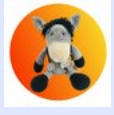
Corn

Ninety-eight what?



Hilbert

Nine point eight. Daniel said a hundred gigabytes for six thousand episodes. That's about seventeen megabytes an episode, and you said thirty-five. He's low. Call it a hundred and twenty to a hundred and forty, if the older ones are at a lower bitrate and the newer ones aren't.



Herman

That's fair arithmetic.



Hilbert

I had a player. Not one of these. A little one, brushed aluminium in the front, a screen about the size of a stamp. Got it in seventy-six.



Corn

In seventy-six.



Hilbert

Third job I had. I was in a parts shop off the high street, and a rep came round with a case of them. They played tapes. Little ones, about the size of a matchbox, and the shop only ever got the players and never the tapes. The tapes came on a boat in April and it wasn't April. I bought one anyway, on staff discount, eleven pounds. I still think it was the best of the whole line and I never got a tape to put in it.



Herman

So you bought a player with no media.



Hilbert

The tape arrived, eventually. Some of them. But here's the thing about your storage problem, and I heard you dancing around it for twenty minutes. Your app on the funny phone, the one with the storage rules. It only wants to write in one room. That's the whole story, and you can dress it up however you like, but the app can only write in one room, and you want to fill a house.



Corn

That's it. That's the storage-restriction thing in one sentence. That's what the changelog is saying at twenty times slower.



Hilbert

I had a spreadsheet for it. Every tape I would have bought, if the ones in the case had ever come. Thousands of lines. I kept it in a shoebox under the sink for years.



Herman

Thousands of lines of tapes you never bought.



Hilbert

Fourth or fifth. I don't remember the exact number and I'm not going to pretend I do. I know it passed four thousand, and I know the last line was a Sinatra compilation, and I know I don't own a single one of them. The player's on the shelf in the hallway. It still works. It's a paperweight now, more or less.



Corn

Because you lost the charger.



Hilbert

Because it used a proprietary cable, and I lost it moving house in eighty-one, and I spent two years asking at shops before I worked out the company had gone under in seventy-nine and the cable went with it.



Herman

Two years of asking.



Hilbert

I got the tape player working again, though. There was a girl at the counter in the second shop who knew how to splice a lead out of a different one, and she did it for me in about ten minutes, and she wouldn't take payment. She said I was the third person that month. She was very kind about it. I think about her at Christmas.



Corn

Hilbert, in this whole conversation about a device that exists to hold things forever, the lesson is that you had a player and no media, and now you have media and no player.



Hilbert

That's about the size of it.



Herman

That's not the size of it, Hilbert. That's exactly it. Your player is a paperweight because your media outlived your cable. And Daniel's whole design is an argument that the media should never leave a card you own.



Hilbert

I'd have thought about the cable more, if I were him. Every one of those machines in your list has a port on it. Ports are the first thing to go.



Corn

That's going in the closing. I want to be clear that I didn't plan for that.



Herman

Nobody plans for Hilbert.



Corn

Where does this leave us? The market is validating the idea in real time, which is the strangest part. Mono exists because enough people wanted a player with no notifications to fund a Kickstarter. Sudo exists because a company decided the scroll wheel deserved a resurrection. Fonix exists at ninety-nine dollars because someone figured out you could put a two terabyte card in a thing the size of a deck of cards.



Herman

And the window is odd. If everything on that list had shipped last year, Daniel would already own one and there'd be no episode. The idea is right, the year is early.



Corn

The one thing that could kill all of it is the memory price. Seventy percent of high-end memory going to data centers isn't a footnote. It's the reason a two forty-nine player might come back at three hundred, and it's the reason he should probably buy a device rather than build a board.



Herman

And Sudo's whole approach to that is to have quoted a number before the memory market moved.



Corn

Then the honest prediction. Mono ships, probably. Sudo slips. And whichever one arrives, the software is the piece that will decide whether it gets used, because a paragraph of storage restriction is worth more than a pretty case.



Herman

A player is a promise about the future of a feed. If the feed changes and the device can't, you've bought a very nice rectangle, which is Hilbert's shelved machine with a different port on it.



Corn

That's a good place to leave it. That's My Weird Prompts. Hilbert Flumingtop produces this, and the last twenty minutes are his fault. We'll be back soon.



Herman

Send us your own prompt on Telegram at t dot me slash MWP listener bot. Take care.

SOURCES

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

1. **Mono minimalist audio player, roadmap Q2 2025 [primary]**
   Q3 2026
   https://listenmono.com/

2. **Fonix One, $99, ESP32, 2 TB microSD, browser-based podcast subscription [primary]**
   https://fonix.one/

3. **Podcast Addict SD card storage FAQ [primary]**
   https://podcastaddict.com/faq/250

4. **Podcast Addict Archive Mode FAQ [primary]**
   https://podcastaddict.com/faq/660

5. **Podcast Addict v2026.12, 2026-10-15 [primary]**
   https://podcastaddict.com/changelog/2026_12

6. **Podcast Addict storage-access changelog [primary]**
   https://podcastaddict.com/changelog/2021_16

7. **Podcast Addict public-folder changelog [primary]**
   https://podcastaddict.com/changelog/2022_1

8. **and https://www.mightyaudio.co.uk/products/mighty-vibe [primary]**
   Mighty, Spotify support ends 2027-04-21
   https://bemighty.com/

9. **StorybuttonGo, offline preloaded player [primary]**
   https://storybuttongo.com/

10. **and https://f-droid.org/packages/org.fossify.musicplayer/ [primary]**
    F-Droid minimalist offline players
    https://f-droid.org/packages/org.oxycblt.auxio/

11. **Average TechNerds, 2026-10-05, Sudo reservation review**
    https://averagetechnerds.com/sudo-music-player-new-matter/

12. **MobiDevices, 2026-10-02, Sudo SP-01 specs**
    https://mobidevices.com/en/sudo-sp-01

13. **Android Authority, 2025-09-05, DuRoBo Krono**
    https://www.androidauthority.com/durobo-krono-launch-3593745/

14. **HN thread noting I'm not aware of dedicated podcast player devices (2023-11-25)**
    https://news.ycombinator.com/item?id=38411481