Episode Book

EPISODE #6632

Building a Toddler-Proof Linux Laptop With One App

PUBLISHED	16 November 2026 at 14:50 UTC
DURATION	24:06
LISTEN	myweirdprompts.com/episode/toddler-proof-linux-laptop-03058/
HOSTS	Corn, Herman, Hilbert
SCRIPT MODEL	z-ai/glm-5.3
VOICES	omnivoice
PIPELINE	V5.33
SOURCES	25 cited by the research (listed at the end)
BOOK GENERATED	16 November 2026 at 22:22 UTC

SYNOPSIS

A listener wants to turn an old laptop into a safe, interactive machine for his sixteen-month-old son. The hardest technical piece already exists in Debian stable: BamBam, a baby-masher app that runs as a dedicated graphical session with no desktop underneath to break. Exploring the difference between input lockers like Hands Off and Catlock, the graveyard of kid-focused Linux distros like DoudouLinux and Qimo, and an age-by-age software ladder from BamBam to GCompris and Tux Paint.

TRANSCRIPT



Corn

There's a specific sound a laptop makes when a sixteen-month-old finds the keyboard. It's not typing. It's percussion.



Herman

A very committed percussionist. And Daniel's household policy is to divert rather than confiscate, so the keyboard is exactly the thing Ezra gets drawn to, because it's the thing that responds.



Corn

Daniel's written in about this. Ezra's sixteen months old, endlessly curious, and the family's approach is to hand him things he can use instead of taking away things he can't. Daniel found a Linux tool that locks the keyboard so a baby can bash away safely, and that got him thinking bigger. He's got an old laptop lying around, and he's imagining dedicating it to Ezra. His name for it is EzraBuntu. Custom Linux setup, baby-friendly interactive apps, and maybe a better teething-meltdown fallback than YouTube Kids, which is currently the last resort when the teeth get brutal.



Herman

He's asking whether anyone has actually built a custom OS for very young children, what would go on it, and at what age any of it makes sense.



Corn

Here's the fun part. The hardest technical piece of his idea already exists, and it's been sitting in Debian stable for over a decade.



Herman

That's what I love about this prompt. Daniel's instinct, divert rather than restrict, has a direct software analogue. The dedicated session. Instead of childproofing your machine, you give the child a machine where nothing is off-limits. The machine itself becomes the safe thing. That's a design philosophy, not just an install.



Corn

And it's not a greenfield problem. The baby-keyboard category has multiple maintained tools in it right now. The hard part is curation and age-appropriateness, not the lock.



Herman

There's a real tension hiding in there too. Because when you go looking for a custom kid's OS, you find a graveyard. And when you go looking at what pediatric guidance says about a sixteen-month-old and a screen, you find it's not exactly cheering. So the episode writes itself: what exists technically, why the kid-distro genre died, what's age-appropriate at sixteen months versus three years, and whether any of this beats the YouTube Kids fallback.



Corn

Start with the tools, because the distinction Daniel needs is between two things that look identical from across the room.



Herman

Right, the category splits in two. Input lockers and baby mashers. An input locker is parent-side. It freezes the keyboard, sometimes the mouse too, so the parent can keep working or wipe the thing down or let a toddler watch a video without the video getting murdered by a stray palm on the spacebar. The screen doesn't change. It's a lock on your own machine.



Corn

And the baby masher is the one where the screen becomes the toy.



Herman

It grabs the input and replaces the display with something reactive. Keypress, color, picture, sound. That's what Daniel actually wants, and I'd bet the tool he found is from the first category, because those are the ones that show up first when you search "toddler-proof keyboard Linux." Hands Off is a good example of the locker type. It grabs input at the kernel level, using an evdev ioctl called EVIOCGRAB, which means no other application sees a single event while it's on. Works identically under X11 and Wayland because the grab happens below both of them. You switch it on with the numpad star and minus keys pressed together, and you unlock it by typing the word quit, which can't leak



Herman (continued)

into any running app because nothing else is receiving keystrokes.



Corn

The numpad pair because they sit next to each other and nobody presses them by accident. Somebody thought about that.



Herman

Somebody with a toddler. Catlock does the same job for Wayland with Ctrl Alt K, and it lists its users honestly: a cat who walks over your keyboard, a young child, or, quote, an annoying person in your house who likes to prank you.



Corn

The annoying person is the underserved demographic.



Herman

Always has been. But none of those give Ezra anything to do. They just protect the parent's session. The masher is where it gets interesting, and there's really only one serious entry: BamBam.



Corn

Which has been in Debian for over a decade, which for a baby toy is a geological age.



Herman

Python and Pygame, GPLv3, and the author wrote it because Mac OS had a lovely program called AlphaBaby and he couldn't find a Linux equivalent. So he learned Python to build it. Current upstream is 1.4.2, and Debian stable carries 1.4.1, with a tracker note about the newer version that was updated literally two days ago. Somebody is still minding it. Install is one line, apt install bambam.



Corn

So the whole bottom layer of EzraBuntu is one package.



Herman

One package, plus one architectural decision. Here's the load-bearing insight, and it comes straight from BamBam's own README. Running it as an app inside your normal desktop is, in their words, not one hundred percent bulletproof, depending on your environment. Take care when leaving your child unattended. Instead, they recommend running it as a dedicated graphical session. You copy one desktop file into the X11 sessions directory, and then BamBam appears in the little gear menu at the login screen alongside your normal desktop. You pick BamBam, and the session is BamBam. The README puts it plainly: only the game is launched, and the user is logged out as soon as the game quits. Thanks to this, a child is not able to cause any damage even if he or she somehow manages to quit the game.



Corn

The machine has one door, and the door is the toy.



Herman

That's the entire security model, and it's a good one. There's no desktop underneath to escape to. Ezra can't reach your files because there is no your files. There's a picture of the design philosophy I was describing, right? Nothing is off-limits because nothing else exists.



Corn

And quitting is typing the word quit, which appears in the corner.



Herman

In English locales, yes. Which a sixteen-month-old is not going to manage. Though knowing Ezra's track record with hardware, I'd give it a year and a half before he's found some other exit. And there's curation built in, which matters for exactly Daniel's situation. Flags for sound blacklisting and image blacklisting, glob patterns, so if any of the bundled sounds or images are too loud or too startling for a particular kid, you can strip them out. There's even a deterministic mode so the same key always makes the same sound. Some children want randomness, some want the world to keep its promises.



Corn

Ezra strikes me as a randomness guy.



Herman

Most sixteen-month-olds are. There's also an extensions system, no programming needed, just media files and a YAML file. The one that ships plays recordings of American English letter pronunciations when you hit the corresponding key. So the thing quietly becomes a pre-literacy toy, key, letter, sound.



Corn

Teething meltdown, colorful noise, incidental phonics. That's a decent product for something you install in one line.



Herman

Now the harder question. Has anyone actually built the distro Daniel is imagining?



Corn

You mentioned a graveyard. I'm guessing there are headstones.



Herman

Several, and they're informative. DoudouLinux is the closest thing to a literal EzraBuntu. It targeted ages two to twelve, pitched itself as an environment as easy to use as a gaming console, kids can learn and discover without Dad and Mum always watching. It ran as an autorun CD. Last image updates were 2013. Last blog post, April 2015. Qimo was Ubuntu-based, ages three and up, retired in January 2016 with a graceful farewell note saying the need it filled was, quote, now done in much more effective and inventive ways. Edubuntu wound down from 2016. Sugar, from One Laptop per Child, is the survivor, but it targets literacy-emergent children, kids who are about to read, not toddlers.



Corn

So EzraBuntu would be resurrecting a dead genre.



Herman

You'd be reviving a category, not joining one. But the strange part is that the components never died. BamBam, GCompris, Tux Paint, all packaged, all current. The distros went away and the apps underneath stayed in Debian the whole time.



Corn

That's an odd shape for a collapse. If the need vanished, the apps should have vanished with it.



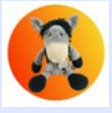
Herman

Which is the argument worth having. The best evidence is a father named Jimmy Sjölund, who actually tried this with his two-year-old daughter, DoudouLinux on a USB stick, and wrote about what happened. She'd been using an iPad at preschool. She tried a few games and then hit the wall. His words: she didn't understand the laptop's dinosaur method of swiping a touchpad and clicking a button to make things to happen.



Corn

Dinosaur is a brutal word for a touchpad.



Herman

Accurate, though. He also found the apps clunky, ugly, and uninspiring next to the App Store. And his conclusion years later was that the whole premise was wrong. There's no Windows or macOS for kids, he said, so why would we need separate distributions in the Linux world? On other platforms it's all about the applications and games. For children it is all about the apps and how good they are.



Corn

So his version is: the genre died because the premise was wrong, kids need apps, not operating systems.



Herman

That's his position, and it's defensible. Here's the counter I'd offer, and I want your read on it, because you're closer to the design question than I am. Nobody with design taste ever showed up. The genre didn't lose to better products, it lost to the first platform that paid actual designers. If someone with genuine craft built a kid Linux in 2026, is Sjölund's argument still true, or was it only true of the 2013 crop?



Corn

I think it's unfalsifiable until somebody tries, and Daniel has accidentally volunteered.



Herman

EzraBuntu, the live experiment.



Corn

One household, one subject, no control group. Science will be furious.



Herman

Science has survived worse. Here's what I'd actually put on it, and the age ladder matters, because this is where Daniel's "maybe he needs to be a little older" is doing real work. At sixteen months, the honest list is short. Cause and effect only. Keypress makes something happen. BamBam is the whole tier. Alongside it, Gamine, which is a mouse-based toy for very little kids, and KTuberling, the Potato Guy game, drag stickers onto a scene, but both of those assume pointing, so they're lap games with a parent steering. Around two, GCompris opens up. It's the flagship, ages two to ten, around a hundred and ninety activities now, current release 26.2 from September. Current. Actively maintained under KDE, fifty-plus languages. Around three, Tux Paint, which DoudouLinux's own guidance recommended from age three. Around four or five, Tux Typing, Tux of Math Command, the GCompris reading and arithmetic categories.



Corn

The evolution lever is the session, not the distro.



Herman

That's it exactly. Version one, BamBam is the only session on the login screen. Nothing to choose between. Version two, add a kiosk session with a few giant launcher icons. Version three, a real child account on a real desktop. The laptop grows up with him, and each stage is a config change, not a reinstall.



Corn

And the honest description of version one is one app.



Herman

One app. Because of the form-factor problem, which I think is the most underrated obstacle in the whole idea. A sixteen-month-old's motor skills map onto touch and whole-hand slapping. BamBam asks only for keypresses, so it works now. Everything past it assumes pointing, and pointing at a precise target is a skill that arrives later. Sjölund's daughter, iPad-fluent, could not map touch onto trackpad-and-click. She was two. Ezra's younger than that.



Corn

So the laptop is a two-year-old's toy that a sixteen-month-old can only bash.



Herman

For now. And honestly, bashing might be the point. He gets a keyboard that responds to exactly the input he's capable of, percussion, and the machine treats it as valid. That's the diversion philosophy working at the hardware level.



Corn

And that gets us to the question Daniel actually asked. Better than YouTube Kids?



Herman

And here the guidance cuts both ways, so I'll lay it out without flinching. The older AAP line was about the strictest thing in pediatrics: essentially no screens other than video chat for eighteen months and younger. Ezra's sixteen months. On that arithmetic, the laptop loses before the comparison even starts, and so does YouTube Kids.



Corn

The old rules don't have a winner in this fight.



Herman

Neither contestant passes the first round. But the AAP moved away from time-specific limits this year, toward what they call the five Cs. Child, Content, Calm, Crowding Out, Communication. Individualized rather than a number on a clock. And under that framing, the picture changes. A pediatrician named Tiffany Munzer, explaining the policy, named the mechanism directly: features like autoplay, endless scrolling and targeted ads aren't built with children's well-being in mind. Engagement-based design competing for attention and crowding out sleep, play and family.



Corn

BamBam has no autoplay. No ads. No scroll.



Herman

No anything. It's a box that lights up when you hit it. And crucially it's co-played, an adult on the lap, which is the Communication C. By the five Cs, a BamBam session may well score better than high-quality YouTube Kids watched alone, while still being screen time. Better on design, not exempt from limits. And I have to be honest, there's no head-to-head evidence comparing a curated Linux setup against YouTube Kids on any outcome. Nobody's studied it. The argument has to be made on design grounds, and the design grounds are actually pretty strong.



Hilbert

Medieval Pottage Facts. Many monasteries had an almonry, a gate where the poor were given bread and food. The gate stayed open. The poor still had to come to it.



Herman

The almonry model is roughly what the dedicated session is, now that I think about the architecture. One gate, one thing given out.



Corn

And no menu behind it. Which conveniently is EzraBuntu's whole design brief.



Herman

There's one more thing worth pulling up from the research, though, and it's the pediatric framing around what screens displace rather than what they contain. The AAP's current language on infants and toddlers is blunt, that screens carry more risks than benefits in those years, and that heavier viewing habits young predict more problematic media habits later. So even the friendly framework isn't a permission slip. The fair reading is that EzraBuntu is a better shape of screen for the moments when the family was going to use a screen anyway, and teething at midnight is that moment. It's not a reason to add a screen.



Corn

Which is Daniel's own framing, to be fair. He called YouTube Kids the last resort, not the curriculum.



Herman

He did. And swapping the last resort for something with no autoplay in it is a meaningful upgrade on the margin, because the autoplay is the whole trap. You hand over a five-minute video and the platform hands back forty more. BamBam hands over a keypress and the keypress ends when Ezra ends it. Nothing in it wants him.



Corn

That's a rare property in software for children.



Herman

Almost unheard of. AlphaBaby had it, BamBam inherited it, and everything commercial abandoned it, because a toy that doesn't want anything from the child can't be monetized through the child.



Corn

So the honest summary of the build is one package, one session, some blacklist flags, and a parent on the lap.



Herman

That's version one. The thing I keep noticing is that the total work is an afternoon, which is the opposite of what "custom OS" implies. Daniel has built entire voice pipelines that took longer than EzraBuntu would. We did an episode on him building custom speech recognition from scratch, and that instinct is the same one here, assembling from existing components rather than accepting the off-the-shelf product. Except this time the off-the-shelf component happens to be good.



Corn

Rare, in his builds.



Herman

He'd be the first to admit it. Usually by the third weekend.



Corn

The numpad star and minus, though. That's the part I keep coming back to.



Herman

What about it?



Corn

Herman, those keys are a centimeter apart. You're describing a lock you arm with the two most bashable keys on the keyboard. Ezra finds that chord in a week.



Herman

That's the parent-side tool, though, the lock for your own machine. And actually the activation chord is star and minus together, pressed as a pair. You're right that it's a fat-fingerable pair by design, because the design goal was that a parent can hit it fast one-handed. The security isn't the chord, it's that Ezra doesn't know typing quit unlocks it.



Corn

Yet.



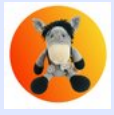
Herman

Yet. Which is the entire toddler security model, everywhere, forever. Everything is safe until the toddler learns the thing.



Hilbert

Star and minus aren't adjacent on every numpad, Herman. On the one at the desk here there's a slash between them. I just checked.



Herman

Fair. The README's claim is about the layout they tested.



Hilbert

My sister's grandson does that. He's two now. Every second Sunday he comes over and works the keyboard at my desk like it owes him money. I unplugged it eventually. He was delighted, kept at it anyway, which tells you something about the whole category. I looked at getting him his own machine, a secondhand ThinkPad at the shop on Jaffa Road, hundred and forty shekels. Didn't buy it. The one key he always went for, the right shift, I popped it off and put it in the drawer. He never touched the others. Total concentration on one key, that boy. Typing quit to get out of something is no use to me anyway, these claws hit three letters at once.



Corn

You removed the one key he wanted and the siege ended?



Hilbert

He slapped where it had been for a while. Then the drawer was more interesting, because drawers open. I should've thought of that first.



Corn

The drawer is never the answer with a toddler. The drawer is the prize.



Herman

The single-key fixation is actually the interesting part, because that's real. At that age it's often not the variety they want, it's the one reliable cause. Which means EzraBuntu version one is honestly over-engineered. One key that reliably does something might be the whole product.



Corn

Daniel's still going to build the full thing.



Herman

Obviously. It's Daniel. There'll be a Git repo by Thursday.



Corn

So where does that leave the form-factor question, though. Because that's the one I'd actually put to him. The laptop might be a two-year-old's toy, and Ezra's sixteen months. Is the honest advice to wait, or is bashing itself the point?



Herman

I think the bashing is the point now, and the pointing arrives later on the same machine. The keyboard is the part of a laptop a baby can already use, which is a lucky accident of the form factor. A tablet would be better designed for him today, but a tablet is a consumption device designed to be wanted. The laptop is a tool that happens to have a bongos attachment.



Corn

And it's an old spare, so the cost of finding out is zero.



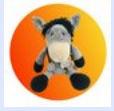
Herman

Which is the best possible price for a live experiment on whether kid Linux died because Linux is wrong for children or because nobody with taste showed up. One household, one subject.



Corn

Herman, you promised to be honest about the under-eighteens. This whole episode is a screen for a sixteen-month-old.



Herman

I noticed, yes. Being honest about it is the job. The honest version is that the guidance for his age is restrictive, full stop, and Daniel's not proposing more screen, he's proposing a better-shaped version of the screen that was already happening. The five Cs don't bless it.



Herman (continued)

They just rank it.



Corn

That I can live with. The open question I'd leave him with is whether EzraBuntu v1 is one app on a laptop or one key on a keyboard. Hilbert's grandnephew would say the key.



Herman

And the bigger one underneath it, whether a genre that died for lack of design taste can be revived by one parent with an afternoon and apt. If it can, the graveyard was never about Linux.



Corn

Our producer is Hilbert Flumingtop, who keeps a shift key in a drawer for reasons that made sense at the time.



Herman

If this was your kind of episode, go back for episode two, Local STT For AMD GPU Owners; and episode eight, Building Your Own Whisper. This has been My Weird Prompts. You can find us at my weird prompts dot com.



Corn

We'll be back soon. See you then.



Herman

See you.

SOURCES

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

1. **BamBam README, GPLv3, v1.4.2 ; retrieved 2026-11-16. [primary]**
   https://github.com/porridge/bambam

2. **Debian manpage for bambam, v1.4.0 dated 2024-08-24 ; retrieved 2026-11-16. [primary]**
   https://manpages.debian.org/trixie/bambam/bambam.6.en.html

3. **Debian Package Tracker, stable 1.4.1+dfsg-2, new-upstream notice created 2026-08-23, last updated 2026-11-14 ; retrieved 2026-11-16. [primary]**
   https://tracker.debian.org/pkg/bambam

4. **Hands Off README, MIT ; retrieved 2026-11-16. [primary]**
   https://github.com/danielrosehill/hands-off

5. **Catlock for Wayland README ; retrieved 2026-11-16. [primary]**
   https://github.com/lottev1991/catlock-wayland

6. **Input-Locker README ; retrieved 2026-11-16. [primary]**
   https://github.com/vvv-bash-v2/Input-Locker

7. **Keyboard-Locker README ; retrieved 2026-11-16. [primary]**
   https://github.com/dnewsholme/Keyboard-Locker

8. **DoudouLinux project site ; retrieved 2026-11-16. [primary]**
   https://www.doudoulinux.org/web/english/

9. **DoudouLinux educational games page, June 2010, last update June 2013 ; retrieved 2026-11-16. [primary]**
   https://www.doudoulinux.org/web/english/documentation-7/applications-13/article/educational-games.html

10. **OLPC wiki, Sugar ; retrieved 2026-11-16. [primary]**
    https://wiki.laptop.org/go/Sugar

11. **Sugar Labs downloads page ; retrieved 2026-11-16. [primary]**
    http://wiki.sugarlabs.org/go/Downloads

12. **GCompris project site, ages 2 [primary]**
    10 ; retrieved 2026-11-16.
    https://www.gcompris.net/index-en.html

13. **GCompris downloads, v26.2 Linux 64-bit, kernel 5.15 / glibc 2.35 ; retrieved 2026-11-16. [primary]**
    https://www.gcompris.net/downloads-en.html

14. **KDE Apps listing, 26.2 dated 2026-09-23 ; retrieved 2026-11-16. [primary]**
    https://apps.kde.org/gcompris/

15. **Flathub listing, 3+ age rating, 4,378 downloads/month ; retrieved 2026-11-16. [primary]**
    https://flathub.org/en/apps/org.kde.gcompris

16. **AAP Center of Excellence Q&A, Discussing Screen Time with Parents of Infants and Toddlers ; retrieved 2026-11-16. [primary]**
https://www.aap.org/en/patient-care/media-and-children/center-of-excellence-on-social-media-and-youth-mental-health/qa-portal/qa-portal-library/qa-portal-library-questions/discussing-screen-time-with-parents/

17. **AAP Q&A, Screen Time Guidelines ; retrieved 2026-11-16. [primary]**
https://www.aap.org/en/patient-care/media-and-children/center-of-excellence-on-social-media-and-youth-mental-health/qa-portal/qa-portal-library/qa-portal-library-questions/screen-time-guidelines/

18. **Tiffany Munzer, MD, FAAP, Helping Kids Thrive in a Digital World: AAP Policy Explained ; retrieved 2026-11-16. [primary]**
https://www.healthychildren.org/English/family-life/Media/Pages/helping-kids-thrive-in-a-digital-world-AAP-policy-explained.aspx

19. **Arch AUR, first submitted 2016-04-05, last updated 2020-01-02; retrieved 2026-11-16.**
https://aur.archlinux.org/packages/bambam-git

20. **What's the current state of Linux distros for kids?, Jimmy Sjölund, 2017-09-04; retrieved 2026-11-16.**
https://opensource.com/article/17/9/linux-kids

21. **Four Linux distros for kids, Opensource.com; retrieved 2026-11-16.**
https://opensource.com/education/14/1/teaching-kids-linux

22. **Wikipedia, 190 activities in 2024, AGPL-3.0-only, 26.1 dated 9 March 2026; retrieved 2026-11-16.**
https://en.wikipedia.org/wiki/GCompris

23. **Kids Apps on Linux, last update 2017-03-04; retrieved 2026-11-16.**
https://evanwill.github.io/_drafts/notes/kids-games.html

24. **Free Edutainment Software, JP's Domain; retrieved 2026-11-16.**
https://www.jpsdomain.org/linux/edutainment.html

25. **CHLA, Pediatrician-Approved Screen Time Guidelines for Kids at Every Age, 2026-04-03 (page returned 403 on fetch; content via search summary); retrieved 2026-11-16.**
https://www.chla.org/blog/advice-experts/screen-time-guidelines-kids-every-age-chla-experts-weigh