Episode Book

EPISODE #5844

Five of the Buggiest Software Ever Shipped

PUBLISHED	10 October 2026 at 19:34 UTC
DURATION	25:05
LISTEN	myweirdprompts.com/episode/buggiest-software-ever-shipped-10023/
HOSTS	Corn, Herman
SCRIPT MODEL	deepseek-flash
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.3
SOURCES	10 cited by the research (listed at the end)
BOOK GENERATED	19 October 2026 at 17:40 UTC

SYNOPSIS

Some software ships broken and stays broken. This episode works through five of the most spectacularly bug-ridden releases ever, separating documented failures from internet folklore and bad design from actual defects. From Windows ME's architecture buckling under itself to a truck racing game that shipped in what its own producer believed was a pre-alpha state, we look at what each was supposed to do, why it failed, how users coped, and whether anyone ever fixed it. Along the way: a reviewer who was told the glitches wouldn't ship, a recovery feature that broke recovery, and a trophy that simply reads "YOU'RE WINNER." At the end, a verdict on which one earns the dubious honour of being the buggiest software ever shipped.

TRANSCRIPT



Corn

There's a particular kind of software where the bug isn't an interruption. It's the product. You don't use it despite the crashes. The crashes are what you're using.



Herman

And you learn the workarounds before you learn the features.



Corn

Daniel sent us a prompt asking for five of the most spectacularly bug-ridden pieces of software ever released. Not good software with one disastrous patch. Not a rough launch that got smoothed over. Products where the bugs were so pervasive, so persistent, and so absurd that they became the user experience. Crashes, phantom data, elaborate workarounds, all just accepted as normal.



Herman

He wants the full picture for each one. What was it supposed to do. Why was it unreliable. The most extraordinary documented failures. How users coped. Whether anyone ever fixed it. And whether any of them became beloved anyway.



Corn

He wants both proprietary and open-source. Operating systems, games, obscure software. He specifically asked us to separate documented failures from internet folklore, and to not confuse bad design with actual bugs. And he wants a verdict at the end. Which one earns the dubious honour of being the buggiest software ever shipped.



Corn

It is. Because the moment you try to rank them, you have to decide what you're ranking. Frequency of failure? Scale of harm? How long it stayed broken?



Herman

Which is why the bad-design-versus-bugs distinction matters so much here. And Windows ME is the perfect place to start, because it's the one everybody names, and it's arguably the one that wasn't really a bug problem at all.



Corn

Explain.



Herman

Windows ME was the last version of Windows built on top of MS-DOS. Everything after it, XP onward, moved to the NT kernel. ME was the end of the line for the old architecture, and it was carrying both sixteen-bit and thirty-two-bit code at the same time, layered on each other. So a lot of what people experienced as instability wasn't a coding defect. It was the architecture buckling under itself.



Corn

So when people say Windows ME crashed constantly, they're describing a design failure.



Herman

Largely, yes. There were real bugs. Documented ones. But the foundation was the problem. And Microsoft made it worse with one specific decision. To make it boot faster, they restricted real-mode DOS. Which sounds like a technical housekeeping detail until you understand what real-mode DOS was for.



Corn

It was the escape hatch.



Herman

Technicians used to boot into DOS to troubleshoot, to replace a corrupt driver, to get at the filesystem when Windows wouldn't start. ME took that away. So you had a less stable operating system and fewer tools to recover it.



Corn

You removed the ladder and then made the floor more slippery.



Herman

That's the whole product.



Corn

What are the documented bugs, though? The real ones.



Herman

There are two big ones. First, System Restore. That was the flagship recovery feature, the thing that was supposed to save you. It turned out that snapshots taken after a certain date were being mis-date-stamped, which broke the recovery. Microsoft patched that one. Then there's a memory issue. Systems with more than five hundred and twelve megabytes of RAM could lose stability, and once you got past about one and a half gigabytes you could hit continuous reboot loops. That's documented under Microsoft's own knowledge base entry, KB three zero four nine four three.



Corn

Continuous reboot loops. So the machine wasn't just crashing. It was catching fire in a loop you couldn't interrupt.



Herman

And this is a home user in two thousand, who has just bought more RAM because more RAM is supposed to help.



Corn

The upgrade makes it worse. That's perverse.



Herman

PC World put it fourth in their Worst Tech Products of All Time list and gave it the nickname everyone remembers. The Mistake Edition.



Corn

Which only works in English.



Herman

The Dutch did better. Over there it was Windows Meer Ellende. More misery.



Corn

That's not a nickname. That's a review.



Herman

Here's the part I find most interesting, though. It was only on sale for four hundred and two days before XP replaced it. That's barely over a year.



Corn

XP arrived in October of two thousand one.



Herman

Correct. And for a product that short-lived, it generated an enormous amount of institutional memory. Every technician who worked in that window kept a DOS boot floppy. They stopped trusting System Restore entirely. They downgraded machines back to Windows ninety-eight as a matter of routine.



Corn

So users coped by reverting.



Herman

They coped by going backwards. The most reliable fix for the new operating system was the old operating system.



Corn

I want to dwell on that for a second, because it's such a strange shape for a fix. The official upgrade path is forward, and the actual working solution is backward. You buy the new thing, you discover it's worse, and the remedy is to reinstall the thing you already had.



Herman

And that's not a workaround in the normal sense. A workaround is a trick you use inside the software. This is a workaround that consists of not using the software.



Corn

It's an anti-workaround.



Herman

It's the user rejecting the premise. And the fact that it became routine tells you the product failed at the level of its reason to exist.



Corn

And did Microsoft ever fix it?



Herman

Partially. The System Restore bug got patched. The architectural instability never did, because you can't patch an architecture. It was solved by abandoning it. XP moved to the NT kernel and the whole class of problem went away.



Corn

Did it become beloved?



Herman

No. It's remembered with contempt. Though there was a review at the time, ActiveWin, that defended it and claimed you saw far fewer blue screens now. Which is one of those claims that's technically defensible if you measure the wrong thing.



Corn

Fewer blue screens. Just more of the machine rebooting itself in a loop forever.



Herman

Yes.



Corn

If Windows ME was a foundation problem, Sonic the Hedgehog in two thousand six was a different failure entirely. That was a game shipped unfinished because a holiday deadline mattered more than the game.



Herman

This is Sonic oh-six. Sonic Team and Sega, released on the Xbox 360 in November of two thousand six. It was supposed to be a franchise reboot for the fifteenth anniversary. Big deal for them.



Corn

So what went wrong?



Herman

The timeline collapsed. Yuji Naka, the series co-creator, resigned in March of that year. The team got split, because they were also making Sonic and the Secret Rings for the Wii. And Sega insisted it had to ship for the two thousand six holiday season.



Corn

There's always a holiday.



Herman

And the producer, Takashi Iizuka, said it plainly afterwards. We didn't have any time to polish and we were just churning out content as quick as we could.



Corn

That's a confession with a press release attached.



Herman

And it's not even the worst version of it. The reporting is that Sonic Team rushed the final stages, ignoring control problems and bug reports that were coming back from Sega's own quality assurance department.



Corn

They ignored their own testers.



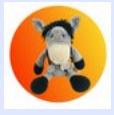
Herman

They didn't have the schedule to act on them.



Corn

Those aren't the same thing, but the result is.



Herman

The result is the same. Metacritic score of forty-six on the Xbox 360, forty-three on the PlayStation 3. Eurogamer called the camera the worst they had ever seen. GameSpot's verdict was that it was a mess from top to bottom.



Corn

And this is Sonic. Sonic is fast movement through complex geometry. The camera is not a nice-to-have in that game. The camera is half the game.



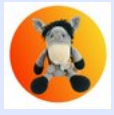
Herman

Which is why the reviews were so harsh. The thing that made Sonic Sonic is the thing that didn't work.



Corn

What's the most extraordinary documented failure?



Herman

I think it's the review saga. Play Magazine, Dave Halverson, gave it a nine point five out of ten. Nine point five.



Corn

For that.



Herman

He was told the glitches wouldn't ship. So he downgraded to eight point five on that understanding. Then the glitches did ship. And when he reviewed the PlayStation 3 version, knowing they hadn't been fixed, he gave it five point five.



Corn

So the reviewer was lied to, and the correction is visible in public.



Herman

In public, in sequence. It's one of the cleaner records of a game shipping with known defects.



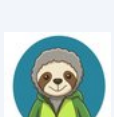
Corn

Did they ever patch it?



Herman

No. The glitches were never fixed. Sega delisted it in twenty ten, and the cited reason was the sub-average Metacritic score. They pulled their own anniversary game from sale because of how it reviewed. It came back on the Xbox 360 marketplace in twenty twenty-two.



Corn

How did users cope?



Herman

They didn't, in the sense of finding a workaround. There's no workaround for a camera. They coped by making it a joke.



Corn

That's a real coping mechanism.



Herman

It's a genuine one. It became one of the most-watched games for glitch showcases. Let's Play channels built whole series out of it. There was a live fandub in twenty nineteen that became its own thing. And the official Sonic account on social media has mocked the game.



Corn

The company joined in on the joke about their own anniversary product.



Herman

Eventually. Once it was old enough to be funny.



Corn

It's worth naming what that coping mechanism actually is, though, because it's not trivial. Turning a broken product into a shared joke is a way of taking ownership of it. The company lost control of the narrative, the players picked it up, and the game's failure became the community's material.



Herman

They converted the defect into culture.



Corn

Which is a very different outcome from what happens when the software is holding something that matters.



Corn

Sonic oh-six was rushed. Big Rigs was something stranger. That one shipped in what its own producer believed was a pre-alpha state.



Herman

Big Rigs, Over the Road Racing. Stellar Stone as developer, GameMill publishing, November two thousand three. It was a truck racing game, sold exclusively through Wal-Mart.



Corn

A truck racing game, sold in the truck aisle.



Herman

Sold exclusively through one retailer. It was outsourced to a Ukrainian team, and the producer, Sergey Titov, has said he believed it shipped in a pre-alpha state. Meaning he did not think it was finished.



Corn

He didn't think it was finished, and it was on shelves.



Herman

On the shelves. Here's the list. There's no collision detection. Trucks pass through opponents and through the scenery. The opponent never moves. The opponent car sits there. You race against a stationary object.



Corn

And you can still lose?



Herman

You can still lose, because there's no time limit, so you can sit there forever. But the opponent does not move. There's no police in the game, despite the box promising police chases. If you select the fourth route, the game crashes. And if you drive in reverse, you accelerate indefinitely.



Corn

Reverse is the fast direction.



Herman

Reverse is faster than forward, and it has no upper limit.



Corn

So the optimal way to play a truck racing game is to drive backwards forever until you exceed the concept of speed.



Herman

And when you win, the trophy reads, in full, all caps, YOU'RE WINNER, space, exclamation mark. With the space before the exclamation mark.



Corn

You're winner. That is the most efficient possible summary of the entire product.



Herman

Metacritic score of six out of one hundred. That's the lowest ever recorded. It started at eight, from five reviews, and then more reviews came in.



Corn

It got worse as more people looked at it.



Herman

GameSpot's Alex Navarro gave it one out of ten and described it as almost completely broken and blatantly unfinished in nearly every way. And then he called it as bad as your mind will allow you to comprehend.



Corn

That's not a review. That's a diagnosis.



Herman

Sergey Titov offered to replace anyone's copy with any Activision Value title. Twenty people took him up on it.



Corn

Twenty.



Herman

Twenty. But here's where it turns. It's got a real cult following. There's a fansite dedicated to it. It was exhibited at the NYU Game Center as part of a show called Bad Is Beautiful. And in April of twenty twenty-five, it was re-released on Steam with a version one point zero patch.



Corn

Twenty-two years later.



Herman

Twenty-two years later, and the patch does one thing. It finally makes the opponent move.



Corn

They fixed the opponent. After twenty-two years.



Herman

That's the patch.



Corn

So did they fix it? Sort of. In twenty twenty-five.



Herman

In twenty twenty-five, partially, and only the one thing that everyone had been laughing about for two decades.



Corn

Windows ME was arguably design. Big Rigs was unambiguous code. That's the distinction Daniel asked us to hold onto.



Herman

Big Rigs is a coding defect from top to bottom. There is no architecture excuse. The collision detection simply isn't there. It was never written.



Corn

Those three are the famous ones. The next two are where this gets darker, and stranger.



Herman

Jet Set Willy. Software Projects, ZX Spectrum, March of nineteen eighty-four.



Corn

Manic Miner's sequel. Manic Miner is one of the canonical British platformers, and Jet Set Willy was the follow-up, and it was the best-selling home game in the UK that year.



Herman

It sold enormously. And on release, it was impossible to complete.



Corn

Not difficult. Impossible.



Herman

Impossible, because of what players named the Attic Bug. There's an arrow sprite in one of the rooms. It travels past the end of the video memory it's supposed to stay inside, and when it goes past the boundary, it starts overwriting the game's own data. Rooms get corrupted permanently. You cannot finish the game.



Corn

The arrow escapes its own room and edits the level.



Herman

That's exactly what it does. And the response from Software Projects is the part that should be taught. Initially, they said it was intentional. The corrupted rooms were poison gas rooms. A feature.



Corn

Poison gas.



Herman

Then they recanted and issued POKEs to fix it.



Corn

A POKE is a direct memory write. On that hardware you could patch a game by typing instructions into memory before running it.



Herman

Publish the fix in a magazine, readers type it in, and the game becomes completable.



Corn

So the fix is distributed by handwriting.



Herman

Via magazine listings. And the completion competition was already running. Two players, Ross Holman and Cameron Else, actually finished it, and they supplied bug fixes as part of it. Cameron Else ended up being hired to port the game.



Corn

The users became the quality assurance team. The winners of the competition were essentially the only functioning QA department the product ever had.



Herman

And that's not the end of it. The BBC Micro port, the Commodore sixty-four port, the Atari eight-bit port, the Dragon thirty-two port. Every one of those was also impossible to complete, for different reasons.



Corn

Different bugs in each version.



Herman

Different reasons entirely. Each port broke in its own way.



Corn

That's the detail that gets me. It's not one bug that got copied across. It's a fresh impossibility on every platform. Which tells you the underlying code was so fragile that the act of moving it to new hardware introduced entirely new ways to fail.



Herman

The same game, broken five different ways.



Corn

And despite being literally unfinishable on release, what happened to it?



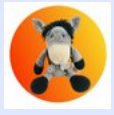
Herman

Retro Gamer readers voted it the sixth best game of all time in two thousand four.



Corn

Sixth best game ever. A game that could not be completed on the hardware it shipped on.



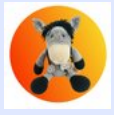
Herman

That's the paradox. The design was good enough that people forgave the fact that it was broken, and rebuilt it themselves.



Corn

Jet Set Willy was impossible to finish. Horizon was impossible to survive.



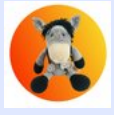
Herman

The Fujitsu Horizon system. Live with the UK Post Office from nineteen ninety-nine to the present. It was point-of-sale accounting software. It tracked the money coming in and out at post office branches.



Corn

And it invented money.



Herman

It invented debts that didn't exist. There are specific, named bugs. The Callendar Square bug, also called the Falkirk bug, created duplicate transaction entries. The Dalmellington bug meant cancelled transactions stayed in play. And the REMM IN bug, where a subpostmaster hitting the previous key to double-check a recorded cash receipt would record it multiple times.



Corn

So looking back to make sure you got it right is what creates the phantom debt.



Herman

Hitting previous to verify. The act of checking is what breaks it.



Corn

That is a cruel bug. The instinct to double-check is the thing that destroys you.



Herman

Justice Peter Fraser found that Horizon contained bugs, errors and defects, and that Legacy Horizon, the version running from two thousand to twenty ten, was not remotely robust. Those are the words of the judgment.



Corn

Not remotely robust. That is a judge being polite and being devastating at the same time.



Herman

David McDonnell, a developer who worked on it, testified that it was beyond anything he'd ever seen even in the twenty-five to thirty years since that project. He said it was a mess.



Corn

Twenty-five to thirty years in software and this was the worst.



Herman

The scale of it. Between nineteen ninety-nine and twenty fifteen, more than nine hundred subpostmasters were wrongfully convicted of theft, fraud, or false accounting, based on Horizon data. Two hundred and thirty-six went to prison. The scandal is linked to at least thirteen suicides.



Corn

Thirteen people dead because the software said the money was missing.



Herman

Because the software said the money was missing, and the software was believed over the people.



Corn

How did users cope?



Herman

That's the problem. They couldn't. There's no workaround for a criminal conviction. The workaround, for years, was to pay back money you didn't owe to make it stop. To accept a shortfall that the system invented, and cover it out of your own pocket, because the alternative was prosecution.



Corn

So the coping mechanism was to absorb the damage personally.



Herman

That's the coping mechanism. And the Post Office spent years insisting the system was robust. Same structure as the poison gas rooms. The system is fine, you're the problem. Except here it ran for two decades and put people in prison.



Corn

Did they fix it?



Herman

The two thousand nineteen High Court case settled for fifty-eight million pounds. The claimants netted about twelve million after costs. Final compensation is expected to exceed a billion pounds.



Corn

Twelve million to the people, out of a fifty-eight million settlement. The costs ate most of it.



Herman

Most of it. And the system is still in use.



Corn

Still in use now.



Herman

Still in use, described as full of patches. It was never replaced.



Corn

And that gets us to the verdict, and I want to do this properly rather than just declaring a winner. There's no actual award for buggiest software ever shipped. No body certifies this.



Herman

It's a rhetorical honour.



Corn

So we should say what we're measuring. Frequency of failure, scale of harm, how long it stayed broken, and whether the bugs were the product itself or a defect within the product.



Herman

Four criteria.



Corn

Four. So let's apply them. Herman, make the case for the one you'd argue.



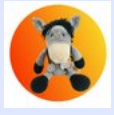
Herman

I'd argue Horizon, and I want to be clear that I'm not doing this for the obvious reason. Anyone can point at a miscarriage of justice and call it the worst.



Corn

Then why?



Herman

Because on the frequency criterion, it wins. Nine hundred and thirty-seven convictions by twenty fifteen. Somewhere around a thousand known cases per year by the end, in thousands of branches, running every day, for over two decades. That is not a bug that happened. That is a bug that was the operating condition.



Corn

And on scale of harm it's not close.



Herman

It's not close. Two hundred and thirty-six people lost their liberty. Thirteen lost their lives. The compensation will cost more than a billion pounds. No video game can compete with that, and I don't think we should pretend otherwise.



Corn

But the criteria pull apart from each other, and that's the interesting part. Because on the fourth criterion, whether the bugs were the product, Big Rigs has a real case.



Herman

Go on.



Corn

The lowest Metacritic score ever recorded. Six out of one hundred. No collision detection. An opponent that doesn't move. A route that crashes. Reverse as the acceleration vector. This is a game where the brokenness is not a flaw within the product. The brokenness is the entirety of the product.



Herman

It shipped in what its own producer called a pre-alpha state.



Corn

Its own producer. And then there's Windows ME, which fails on a different axis. It fails the bug criterion entirely and still makes the list, because the architecture was the defect. It sold for four hundred and two days. Four hundred and two days, Herman.



Herman

And there's no formal calculation for that.



Corn

There isn't. But four hundred and two days and a knowledge base article about reboot loops is a strong résumé.



Herman

You're not going to pick Big Rigs.



Corn

I'm not going to pick Big Rigs, and I'll tell you why. I've been sitting here trying to construct the argument and it keeps collapsing on one point, which is that the harm requirement keeps coming back into the room.



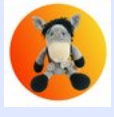
Herman

Say it.



Corn

You can put the controller down. That's the whole difference. With Sonic oh-six, with Big Rigs, with Jet Set Willy, you can put the thing down and walk away and the world continues. With Horizon, there is no putting it down, because the person who would be putting it down is the person who is being prosecuted for theft.



Herman

The user can't exit.



Corn

So Horizon wins, but I want the verdict stated precisely, because it's not a verdict about bugs. On the narrow reading, where I'm ranking sheer quantity and absurdity of defects, it's Big Rigs. On the honest reading, where I'm asking which product's bugs did the most damage, it's Horizon, and there's no second place.



Herman

I'll take that. And there's something underneath all five that I keep coming back to.



Corn

Which is that the same phenomenon produces opposite results depending on what the software was responsible for.



Herman

That's it. A truck racing game that doesn't work is a meme. Accounting software that invents phantom debts is a prison sentence. Same class of defect. Same pervasiveness, same persistence, same absurdity. Totally different consequence.



Corn

And the difference is entirely in what the thing was holding on behalf of its users.



Herman

Entirely. Which means the word "bug" is doing two jobs in this episode. It describes a defect, but it also describes the scope of the thing it was allowed to touch. And those two jobs get confused constantly.



Corn

Which raises the question we haven't answered. Why does the same failure get comedy in one case and catastrophe in the other, and who decides which one you got?



Herman

The person who decided how much power to hand the software.



Herman

Windows ME, Sonic oh-six, Big Rigs, Jet Set Willy, Horizon. Five for five.



Corn

Sonic oh-six, does it hold up as a candidate? I want to press on this one because I think it's the weakest.



Herman

Why?



Corn

Because at least part of the fix was possible. Titov offered replacements. Software Projects issued POKEs. Microsoft patched System Restore. Horizon never actually got fixed. It's still running, full of patches, right now.



Herman

That's the difference you want to make.



Corn

The only bugs that never got fixed were the ones nobody was being prosecuted for. That's backwards. The one that put people in prison is the one that never got replaced. Sonic oh-six sold eight hundred and seventy thousand copies. Half a million units of a ninety-two pound experience and a full price sticker on a game its own team had no time to polish. That is not an obscure failure. That is a failure that went to scale.



Herman

And it never got patched. So the scoreboard is: Big Rigs got its patch twenty-two years later. Windows ME got superseded by XP. Jet Set Willy got POKEs from magazines. Sonic oh-six got nothing. Horizon got fifty-eight million pounds and a billion pounds more to come, and is still doing the same job now.



Corn

That's the whole list.

SOURCES

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

1. **Wikipedia (accessed 2026-10-10)**
   https://en.wikipedia.org/wiki/List_of_software_bugs

2. **Wikipedia (accessed 2026-10-10)**
   https://en.wikipedia.org/wiki/Windows_Me

3. **(2006_video_game)**
   Wikipedia (accessed 2026-10-10)
   https://en.wikipedia.org/wiki/Sonic_the_Hedgehog

4. **Wikipedia (accessed 2026-10-10)**
   https://en.wikipedia.org/wiki/Big_Rigs:_Over_the_Road_Racing

5. **Wikipedia (accessed 2026-10-10)**
   https://en.wikipedia.org/wiki/British_Post_Office_scandal

6. **Communications of the ACM, Mark Halper, 2025-01-08**
   https://cacm.acm.org/news/how-software-bugs-led-to-one-of-the-greatest-miscarriages-of-justice-in-british-history/

7. **Wikipedia (accessed 2026-10-10)**
   https://en.wikipedia.org/wiki/Jet_Set_Willy

8. **Wikipedia (accessed 2026-10-10)**
   https://en.wikipedia.org/wiki/Microsoft_Bob

9. **NPR, 2017-05-31**
   https://www.npr.org/2017/05/31/530235165/total-failure-the-worlds-worst-video-game

10. **Paleotronic Magazine, 2020-10-09**
    https://paleotronic.com/2020/10/09/the-scourge-of-windows-millennium-edition/