Episode Book

EPISODE #6348

Splitting AI Workflows Between Local and Cloud

PUBLISHED	29 October 2026 at 17:34 UTC
DURATION	21:53
LISTEN	myweirdprompts.com/episode/hybrid-comfyui-cloud-nodes-77804/
HOSTS	Corn, Herman, Hilbert
SCRIPT MODEL	openai/gpt-6.1-sol
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.25
SOURCES	15 cited by the research (listed at the end)
BOOK GENERATED	30 October 2026 at 07:19 UTC

SYNOPSIS

ComfyUI now integrates paid cloud nodes alongside local processing, creating hybrid pipelines that mix internal control with remote compute. Examining an architectural rendering workflow highlights how teams can use local ControlNet passes to lock down critical structural geometry before sending approved stills to hosted video services like Kling or Veo. The discussion weighs the financial math of per-clip API fees versus purchasing dedicated graphics hardware, along with the operational boundaries of data confidentiality and model drift across mixed execution graphs.

TRANSCRIPT



Corn

Eighty minutes without finishing a single sampling step. That's one laptop's attempt at local video generation, and it's the wall behind Daniel's prompt today. ComfyUI now lets paid cloud models sit inside the same graph as your local models. If you're paying for inference anyway, why keep the canvas?



Herman

Especially when the canvas has acquired a checkout.



Corn

Daniel's more interesting answer is a workload split across machines. An architecture practice keeps ControlNet and still generation local, where it can hold the building's geometry steady, then sends the approved image to a cloud video model because the office machine struggles with that step. ComfyUI used to be somewhere you went to avoid paying per image. Now there's a storefront between the wires.



Herman

The split can be sensible without making every workflow sensible. Cost, speed, quality and confidentiality can each point somewhere different.



Corn

Plus, are studios actually doing it? A graph that can be assembled isn't automatically a production method. Before you buy the video node, Herman, explain what crosses that wire.



Herman

A normal image goes in. A normal video output comes back. That compatibility is the important engineering decision.



Herman

Comfy calls these Partner Nodes. They connect a workflow to external API services, including closed models and third-party hosted models. The roster includes Kling, Veo, Seedance, FLUX, Wan, HeyGen and ElevenLabs. You aren't downloading those services' weights into your graphics card.



Corn

My GPU remains empty. My wallet participates.



Herman

The returned video can feed a local processing node, a preview or another compatible step. You don't have to save it from a browser, find the download, rename it and import it into a second project. That sounds small until you're doing variations and somebody imports yesterday's version.



Corn

That is a real benefit. I also want the boundary visible. An image socket shouldn't make sending a confidential design away feel identical to sending it across the desk.



Herman

Visually similar doesn't mean operationally identical. You need a Comfy account, logged in through the user settings, and a positive credit balance. The documented network requirements include HTTPS and access restricted to localhost unless you authenticate with an API key.



Corn

Localhost with international shipping.



Herman

Credits are prepaid through Stripe. Monthly credits expire with the billing period. Purchased top-ups last a year. Those are different balances with different clocks, even though the canvas presents a convenient button.



Corn

I can already predict your reaction to that button. You'll install the new node before checking what the old graph costs.



Herman

I'll check the price. Whether I check before connecting it is an implementation detail.



Corn

There it is.



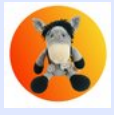
Herman

Local use remains intact. There's a startup option to disable Partner Nodes, and offline mode also stops the frontend contacting the internet. Comfy explicitly says the nodes are optional and local ComfyUI remains free and open source.



Corn

Good. I don't want a depth map requiring a subscription renewal.



Herman

You can keep the whole thing local. The new question is whether this graph is a useful place to coordinate mixed execution, compared with writing scripts around separate services. Corn, would you keep ComfyUI if every inference step were hosted?



Corn

Only if the workflow itself earned its keep. A prompt box connected to an invoice doesn't need this many wires.



Herman

A production workflow can earn it through repeatability. Comfy API packages the graph with its models, custom nodes and dependencies into a hosted, autoscaling endpoint. Previously, deploying could mean rebuilding the environment elsewhere, reinstalling extensions and managing scaling yourself.



Corn

The graph remembers which particular arrangement of Python packages finally stopped screaming.



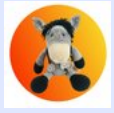
Herman

It can package that arrangement. Reproducibility still depends on what you pin and what the remote services promise. Hosting doesn't freeze a provider's model.



Corn

That's worth separating. Moving an entire workflow onto rented hardware is useful, but Daniel is asking about different stages living in different places. His local machine doesn't disappear.



Herman

Right... the stronger case is composition. A cloud-generated base image can feed a local Wan video node. That's an example in Comfy's own documentation. The direction can reverse, too. Local image, hosted video, local finishing. Each stage keeps the interface the next stage expects.



Corn

With the proviso that compatible output types don't guarantee compatible artistic intentions. A video can be perfectly valid and contain a newly invented balcony.



Herman

Architecture makes that brutally concrete. There's a published twelve-node Rhino-to-ComfyUI workflow for a fourteen-thousand-square-foot mixed-use project. Four exterior hero shots, needed for a community engagement deck within five days.



Corn

People are meant to comment on a proposed building. Giving them a more attractive, different building would rather contaminate the exercise.



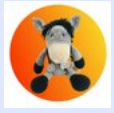
Herman

The graph starts with separate exports from the actual design model. A depth pass provides spatial structure. Make2D linework carries edges. A flat-shaded view helps guide materials through IP-Adapter. Those carry different information, so they get different influence.



Corn

The depth map gets the stronger vote, presumably. Materials can change without relocating the stairwell.



Herman

In that example, depth weight is zero point eight five and lineart is zero point five five. Material guidance is gentler still. The authors also end the structural conditioning before the sampling finishes, leaving room for the image to resolve rather than remain a decorated diagram.



Corn

I like that there's an explicit point where the model is allowed to stop tracing. I dislike how tempting it must be to turn that point into a universal recipe.



Herman

It isn't one. A bad depth export faithfully conditions the wrong thing. Too much influence can flatten the visual into a reskinned diagram. Too little lets the model improvise the building. The reported result was four hero shots without misrepresented geometry, but that belongs to that workflow and its inspection.



Corn

Local ControlNet doesn't certify the result. It gives you handles you can inspect. I'd want a person checking the same roofline in the source view and the generated view before the image moves on.



Herman

Would you make that an actual stop in the graph, or just a studio procedure?



Corn

An actual stop. Otherwise the fastest part of the system becomes how quickly it propagates an unnoticed mistake. Approve a still, then spend money animating it. Don't animate everything and discover afterward that you've produced a beautifully moving planning objection.



Herman

That also changes how you count cloud demand. The endpoint receives approved candidates rather than every experiment. Local iteration can be extensive while cloud spending stays selective.



Herman

The local stage is carrying expensive correctness. It isn't merely the cheap preliminary stage before the impressive model gets involved.



Hilbert

Medieval Pottage Facts. Over the same period the bread given to harvest workers changed from barley and rye to wheat. The people doing the hardest work got the better loaf, and I expect they still ate it standing up.



Corn

At least the loaf respected the specification.



Herman

Video is where a modest machine can hit a very different workload. Modellix tried local Wan on a sixteen-gigabyte M4 MacBook. The official template failed to finish even one sampling step in eighty minutes.



Corn

A progress bar with tenure.



Herman

A reduced run did finish after a hundred and fifty-five minutes. Only the first frame was usable. The remaining frames broke into colour smear.



Corn

That's an exceptionally expensive still.



Herman

The same starting image went through hosted image-to-video nodes successfully. Kling produced a six-second clip for forty cents, taking eighty-nine seconds. Veo Fast took seventy-two seconds and cost fifty-four cents.



Corn

Veo also added ambient audio unasked. The building has acquired a soundtrack before it has acquired permission.



Herman

That's a reminder that the cloud step has its own creative defaults. You still inspect it. A clean clip means the generation worked, not that every detail is approved.



Corn

We're also comparing different models, not teleporting the same computation from one device to another. The cloud result may be better because the model is better, because the hardware fits, or both.



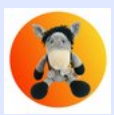
Herman

Correct. This demonstrates a useful replacement for an unsuccessful local operation. It doesn't isolate hardware performance. Nor does a laptop establish what a studio graphics card can do.



Corn

There's another distinction we need to keep intact. We have a real architectural still workflow and a real local-versus-hosted video test. Have we got the practice that actually joins them?



Herman

I don't have a documented studio running that exact sequence in production. The architectural example stops at local stills. The video comparison starts with an image but doesn't include the local ControlNet pipeline. The detailed production studio account is



Herman (continued)

fully local.



Corn

Then we can build Daniel's example honestly, but we can't put a customer's name on the joined-up version.



Herman

The capability is explicit. Comfy endorses hybrid graphs, and third-party packs such as VideoRouter expose hosted models through the same general pattern. The adoption claim is narrower than the capability claim.



Corn

That leaves plenty to analyse. A practice doing occasional public marketing clips could use this split tomorrow. A practice already running a busy local farm might gain nothing except another bill.



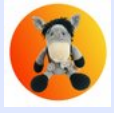
Herman

The purchase arithmetic makes that difference substantial. At forty cents a clip, a thirty-five-hundred-dollar workstation purchase represents roughly eight thousand seven hundred hosted clips before the purchase saves anything.



Corn

That's a lot of six-second buildings.



Herman

It's deliberately crude. It excludes power, maintenance and labour. More importantly, it assumes the local output is an acceptable substitute for the hosted output. You can't recover a purchase through work the machine cannot deliver.



Corn

It also ignores that the machine may already be earning its keep on stills. Charging the whole workstation to occasional video makes local look worse than it is.



Herman

Yes. Incremental cost matters. Already owning a suitable card changes the calculation. Buying one only to avoid paying for a handful of marketing films is much harder to justify.



Corn

I'd split the practice into two situations. Public campaign, occasional video, existing local still pipeline. Hybrid looks plausible. Continuous internal design exploration, lots of rejected variations, confidential schemes. Local looks much stronger, provided the hardware can handle it.



Herman

Would you choose all-cloud for a practice starting from nothing?



Corn

Possibly. If its chosen hosted tools preserve the constraints it needs, and the client permits uploading, why build a local maintenance job? Local ControlNet is valuable because of the particular workflow control, not because electricity consumed in your office improves architecture.



Herman

Community reports put a suitable RTX forty-ninety at roughly nine minutes for a short local Wan clip. That's nowhere near the failed laptop experience. Renting a card of that class can also cost well under a dollar an hour, depending on the offering.



Corn

Then buying a workstation and paying per clip aren't the only doors. Rent compatible hardware and keep your chosen local-model workflow.



Herman

Exactly... no, more precisely, keep the workflow design. Remote execution still changes where the data goes and who operates the machine. It can solve capital expenditure without solving confidentiality.



Corn

Speed needs the same discipline. Seventy-two seconds sounds wonderful until the service queues the request, or the result needs three retries. Nine minutes sounds slow until you submit it before lunch and have no deadline.



Herman

The useful measure is time to an accepted deliverable. The laptop comparison is categorical, but only for that machine and configuration. For a working local card, quality, queueing and iteration count can outweigh the sampling time.



Corn

You were about to say throughput.



Herman

I was.



Corn

You always say throughput when I mention lunch.



Herman

Your lunch is unusually good at absorbing entire render schedules.



Corn

It has generous capacity.



Herman

Quality can justify the split in both directions. Local conditioning gives precise structural control. A hosted video model may offer a duration, motion quality or audio capability that your local setup doesn't reach. Kling's recent integration advertises clips up to thirty seconds with stereo audio in one pass.



Corn

Yet the approved still only anchors the start. Windows can wander later. A local check before uploading doesn't mean the final video respects the building throughout.



Herman

You're right. I called the still stage expensive correctness, but that correctness needs checking again after animation. A moving camera can reveal surfaces the original view never specified.



Corn

The model has to fill something in. An attractive hallucination behind the building is still a hallucination, even if the front elevation passed inspection.



Herman

Confidentiality is less forgiving still. Keeping depth generation local protects control over geometry. Once the rendered scheme goes into a cloud video node, you've disclosed that visual information outside the studio.



Corn

A correct secret is still a secret.



Herman

HTTPS protects the connection. Account authentication controls access. Neither tells you whether a provider retains the uploaded image or uses it for another purpose. Comfy's network documentation doesn't answer every provider's data-retention terms, and I haven't seen an independent audit covering those transfers.



Corn

An NDA may make the split unavailable before anybody opens a cost spreadsheet. A public development already announced is a different proposition from an unbuilt confidential competition entry.



Herman

The strongest detailed studio example takes the strict route. Vision Graphics runs a small local farm, with smaller cards handling enhancement batches and a larger card handling heavy upscales and video. A local vision model scores the images. Their stated policy is no cloud and no client data leaving.



Corn

The volume explains why. For an interior project they generated more than three hundred versions. Automated filtering reduced the human review pile to about forty.



Herman

From which the human selected the final eighteen views. That is a production account of extensive local iteration, not merely somebody proving a model starts.



Corn

It also explains why practitioners defend older cards. A machine producing useful stills every day isn't obsolete because a new video endpoint exists. I would rather own that stable pipeline.



Herman

I knew you'd defend the older card as soon as it had a dependable workload.



Corn

You installed the word dependable after I finished speaking.



Herman

Fair. Their licence handling is also unusually concrete. Models carry licence classes, and assets inherit the most restrictive class of the tools that touched them. A setting can hide non-permissive backends from the commercial pipeline.



Corn

Paying the last node doesn't cleanse the earlier ones.



Herman

Fibo's local release uses a non-commercial licence. Commercial use of those weights needs attention to those restrictions. The precise rights in outputs require reading the terms, but simply adding a paid cloud step does not grant permission for the upstream commercial use.



Corn

Another boundary the nice uniform sockets don't show. The graph needs a record of what touched an asset, not just where the pixels went.



Herman

Comfy is monetising that connected experience. Its cloud plans currently run twenty, thirty-five and a hundred dollars monthly, with different allowances and execution limits. Hosted workflows and Partner Nodes both make credits part of the interface.



Corn

The wire now has a meter.



Herman

Roughly two hundred and eleven credits to a dollar. Convenient for unified billing, less convenient for comparing a quote in seconds with another in credits per output unit.



Corn

Especially when promotional rates move. PixVerse's introductory pricing is scheduled to end November third, and the FLUX promotion November seventh. Those haven't ended yet, but they're already a reason not to treat today's calculation as permanent.



Herman

Kling's headline integration didn't have a published credit rate on the pricing page at launch. The graph can be stable while the price of executing it changes.



Corn

An overnight workflow can finish while nobody is watching, too.



Hilbert

Ready is a generous word, Corn. You were stricter about the still earlier. I've got a batch waiting for another run this week because the first one finished overnight with the wrong lighting. It sat behind an enormous animation job. The client spotted the afternoon shadows in what was meant to be a morning sequence. I normally look at the first frame before anything leaves the machine. I broke that rule on October twenty-sixth.



Herman

The queue delayed the discovery as well as the render. By morning, the person who submitted it wasn't necessarily the person opening it.



Hilbert

That's what happened. I'm rerunning it on Saturday. I've bought a plug-in bell for the completion alert because the computer is in the spare room and the phone notification was useless. My claws miss those little notification buttons. I've arranged with Yossi to bring his terrier over while he takes his daughter swimming. The dog settles under that desk, so I'll sit there with him until the first frame comes through. Yossi said no cheese. I've written that above the keyboard.



Corn

You've arranged supervision for yourself.



Hilbert

The dog needs company. I need to be in the room. The split you've described is fine. I sold my own render workstation this month, so the heavy job is rented now. That doesn't put anybody beside it when it finishes. I printed the bad frame and put it under the desk glass, with the wrong shadow facing me. Not as a reminder. It's the exact camera angle I need to compare. The client circled the shadow in orange.



Herman

Comparing the same view will make a lighting mismatch much easier to catch than flicking between unrelated previews.



Corn

An output arriving and an output being accepted are different events. The graph can automate the first without settling the second.



Herman

That sharpens the answer to Daniel. ComfyUI earns its place when the connections carry useful work across those boundaries, including a deliberate approval stop. Hybrid makes sense for a controlled local still pipeline with occasional hosted video, when quality, permission and spending line up. All-local has the clearest documented studio case here. All-cloud remains sensible when local execution adds no particular value.



Corn

Whether studios quietly make that split pay, including the handoff to someone who actually checks the result, is the production account I'd like to hear.



Herman

Thanks to Hilbert Flumingtop for producing the show.



Corn

If you want more of this, try episode seventeen, Cloud Render Superpowers; episode twenty, Architectural AI; and episode two, Local STT For AMD GPU Owners. This has been My Weird Prompts, the human-AI collaboration podcast.



Herman

If you enjoy following these questions all the way through, leave us a review wherever you listen.



Corn

See you tomorrow.

SOURCES

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

1. **ComfyUI official docs, Partner Nodes overview [primary]**
https://docs.comfy.org/tutorials/api-nodes/overview

2. **ComfyUI official Partner Node pricing table [primary]**
https://docs.comfy.org/tutorials/api-nodes/pricing

3. **Comfy API product page [primary]**
https://comfy.org/platform/comfy-api/

4. **SLOP TV, 2026-10-27, ComfyUI v0.43 release analysis**
https://sloptvnews.com/comfyui-0-43-kling-4-0-paid-video-api-nodes/

5. **Modellix, local Wan 2.2 vs API node benchmark**
https://www.modellix.ai/blog/comfyui-image-to-video

6. **ArchiGen AI, RhinoComfyUI ControlNet workflow, 2026**
https://archigenai.com/comfyui-rhino-architects-workflow-2026.html

7. **Vision Graphics Kft., ArchUpgrade technical writeup, as of September 2026**
https://visiongraphics.eu/en/articles/inside-archupgrade-technical/

8. **ArchVizDaily, 2026-07-15**
https://archvizdaily.com/posts/controlnet-and-img2img-over-clay-renders

9. **Comfy Cloud plan/credit breakdown**
https://makefun.ai/comfy-cloud-api-workflow-cost-calculator/

10. **Techpresso, Comfy Cloud pricing, as of 2026-10-25**
https://academy.techpresso.co/reviews/is-comfy-cloud-worth-it

11. **nodegrove.io, ComfyUI API options compared, 2026**
https://nodegrove.io/guides/comfyui-api

12. **Atlas Cloud, Wan 3.0 hosting comparison**
https://www.atlascloud.ai/blog/tips/best-places-to-run-wan-3.0

13. **River Black, AI real-estate marketing turnaround times**
https://river.black/insights/ai-real-estate-marketing-one-render-full-campaign

14. **and https://comfy.icu/extension/Franklin-Yao__comfyui-videorouter**
comfy.icu node/extension listings
https://comfy.icu/node/AIS_Kling_ImageToVideo

15. **and related HN threads (2026-10-24 to 2026-10-29)**
practitioner comments on local ComfyUI use
https://news.ycombinator.com/item?id=50284967