ComfyUI + LTX-Video 2.3 文生视频：全模型部署 · 适配 · 使用完整报告（V2）

运行平台：AMD Instinct MI308X（192 GB HBM，gfx942） / ROCm 7.2 / 服务器 10.1.29.48

应用：ComfyUI 0.27.0 + 自定义节点 ComfyUI-LTXVideo（容器 `comfyui`，端口 8188）

模型系列：Lightricks LTX-Video 2.3（22B，视频 + 音频生成）

报告日期：2026-07-07（V2：覆盖全部 7 个交付模型，含 gemma-fp8 与 Licon-MSR LoRA 补齐后重跑）

一、概述与结论摘要

本报告是 AMD Instinct MI308X（ROCm）服务器上 **ComfyUI + LTX-Video 2.3 文生视频**工作的完整记录（V2），在前一版基础上**补齐并实跑了全部 7 个交付模型**，含此前缺失的 `gemma_3_12B_it_fp8_e4m3fn`（fp8 文本编码器）与 `LTX-2.3-Licon-MSR-V1`（多主体参考 LoRA）。全部模型经 ComfyUI HTTP API 提交真实工作流、统一参数重跑一遍，采集加载方式、显存、耗时、产物与适配问题。

7 个交付模型清单与覆盖状态（全部 √ 实跑通过）：

#	模型	类别	部署路径	大小	本轮实跑
1	LTX23_video_vae_bf16.safetensors	视频 VAE	`vae/`	1.35 GB	√ 成片解码
2	LTX23_audio_vae_bf16.safetensors	音频 VAE	`vae/`	0.33 GB	√ 编解码往返
3	taeltx2_3.safetensors	快速预览 VAE	`vae/`	23 MB	√ 预览解码
4	gemma_3_12B_it_fp8_e4m3fn.safetensors	fp8 文本编码器	`text_encoders/`	13.2 GB	√ 出片（本次新部署）
5	Singularity-LTX-2.3_OmniCine_V1.safetensors	电影化 LoRA	`loras/`	2.6 GB	√ 出片
6	LTX-2.3-Licon-MSR-V1.safetensors	多主体参考 LoRA	`loras/`	624 MB	√ 出片（本次新部署）
7	ltx-2.3_text_projection_bf16.safetensors	文本投影	`text_encoders/`	2.15 GB	√ 存在/有效/可选（详见 §六）

核心结论：

1. **7 个模型在 AMD / ROCm 上全部成功加载并正确运行**，无 dtype/算子/内核错误。本轮 5 组测试（基线双解码 / Singularity LoRA / Licon-MSR LoRA / 音频往返 / gemma-fp8）全部 `status=success`、`node_errors` 为空。
2. **fp8 文本编码器（gemma_3_12B_it_fp8_e4m3fn）实跑成功**：作为 `LTXAVTextEncoderLoader` 文本编码器槽的直接替换即生效，出片质量与全精度版等价；磁盘体积 13.2 GB，较全精度版（22.7 GB）小约 42%，是"省显存/省加载"的等效替换。
3. **两个 LoRA 均生效且方向明确**：Singularity 重构为电影级人物特写、消除基座烧录字幕；Licon-MSR（多主体参考）出片画面最干净、构图完整，其完整能力（2–5 张参考图→视频）需配套 `ComfyUI-Licon-MSR` 插

件，本次以文生视频（T2V）验证其可加载可生效。

4. **两个易踩适配点：**① 独立音频 VAE 必须用通用 `VAELoader` 加载（专用 `LTXVAudioVAELoader` 只读 `models/checkpoints`）；② 合并版 fp8 checkpoint 是前提（拆分 transformer 缺文本投影会导致 3840/4096 维度不匹配报错）。
5. **模型来源补记：**gemma-fp8 仅 HuggingFace `GitMylo/LTX-2-comfy_gemma_fp8_e4m3fn` 有（ModelScope 无镜像），13.2 GB；Licon-MSR 取自 ModelScope `LiconStudio/LTX-2.3-Multiple-Subject-Reference`。下载 gemma-fp8 时用 8 路 curl range 并行把 hf-mirror 从单线程 4 MB/s 提速到约 17 MB/s。

二、运行环境

项目	配置
GPU	AMD Instinct MI308X ×8（每卡 192 GB HBM，gfx942）；本任务 ComfyUI 固定用 GPU0（`HIP_VISIBLE_DEVICES=0`）
主机内存	约 1.5 TB
GPU 软件栈	ROCm 7.2
运行方式	Docker 容器 `comfyui`（镜像 `vllm/vllm-openai-rocm:v0.20.1` 复用为 ComfyUI 运行时），数据挂载 `/srv2/comfyui`（本地 NVMe）
应用	ComfyUI 0.27.0（前端 1.45.20）+ 自定义节点 ComfyUI-LTXVideo
服务	`main.py --listen 0.0.0.0 --port 8188`
模型根目录	`/srv2/comfyui/ComfyUI/models`（容器内 `/comfyui/ComfyUI/models`）

三、模型与文件清单（部署核查）

除 §一 的 7 个交付模型外，工作流依赖的配套模型（一并核实在位）：

类别	文件	路径	大小	作用
合并 checkpoint	`ltx-2.3-22b-dev-fp8.safetensors`	`checkpoints/`	28 GB	含 transformer + VAE + 文本投影，官方工作流必需
文本编码器（全精度）	`gemma_3_12B_it.safetensors`	`text_encoders/`	22.7 GB	Gemma 3 12B bf16，与 #4 fp8 版等价可互换
蒸馏 LoRA	`ltx_2.3_22b_distilled_1.1_lora_...safetensors`	`loras/`	2.6 GB	少步加速（未启用）
空间超分	`ltx-2.3-spatial-upscaler-x2-1.1.safetensors`	`latent_upscale_models/`	995 MB	二阶段 ×2 放大（未启

				用）

> 模型来源：合并 checkpoint `Lightricks/LTX-2.3-fp8`；全精度文本编码器与投影 `Comfy-Org/ltx-2`；3 个 VAE `Kijai/LTX2.3_comfy`；Singularity LoRA `WarmBloodAban/Singularity-LTX-2.3_OmniCine_V1`；gemma-fp8 **`GitMylo/LTX-2-comfy_gemma_fp8_e4m3fn`**（HuggingFace）；Licon-MSR LoRA **`LiconStudio/LTX-2.3-Multiple-Subject-Reference`**（ModelScope）。本次新部署 gemma-fp8 与 Licon-MSR 两项。

四、部署跑通的关键：从"卡住"到"成功"

4.1 问题定位

初期下载的是拆分版 `ltx-2.3-22b-dev_transformer_only_bf16`（仅 transformer，约 40 GB），用软链伪装成 checkpoint。官方工作流通过 `CheckpointLoaderSimple` 加载**合并** checkpoint，文本投影由 `LTXAVTextEncoderLoader` 连同 Gemma 从该 checkpoint 读取。拆分版缺打包的文本投影 → **文本嵌入维度不匹配（期望 3840、实得 4096）** → 报错。

4.2 解决方案

1. 下载合并版 fp8 checkpoint `ltx-2.3-22b-dev-fp8.safetensors`，删除错误软链。
2. 文本编码器沿用 `gemma_3_12B_it.safetensors`（Comfy 官方打包版）。

4.3 避开的两个坑

- 官方"完整示例工作流"依赖未安装的 RES4LYF 节点（`ClownSampler_Beta`）和需要外部 API Key 的 `GemmaAPITextEncode` → 不使用。
- 改用核心 + LTXVideo 节点手搭最小可用工作流（纯本地、无需 API Key），经 ComfyUI HTTP API 提交，一次通过。

五、工作流与统一测试参数

5.1 最小可用文生视频节点链路

```
CheckpointLoaderSimple(ltx-2.3-22b-dev-fp8)      → MODEL / VAE
LTXAVTextEncoderLoader(gemma + ckpt)             → CLIP(含文本投影)
CLIPTextEncode(正向) ┐
CLIPTextEncode(负向) ┘  → LTXVConditioning(frame_rate=25)
CFGGuider(model, pos, neg, cfg=3.0)
KSamplerSelect(euler) + LTXVScheduler(steps=20) + RandomNoise(seed=101)
SamplerCustomAdvanced(...) → LATENT
VAELoader(独立VAE) → LTXVTiledVAEDecode / VAEDecode → CreateVideo(fps=25) → SaveVideo
```

5.2 本轮统一参数

参数	取值
分辨率	768 × 512
帧数 / 帧率	65 帧 / 25 fps
采样	euler，20 步，CFG=3.0，LTXVScheduler(max_shift 2.05 / base_shift 0.95)，seed=101
视频解码	独立 video VAE 经 `LTXVTiledVAEDecode`（2×2 分块，overlap 6）

提示词	阳光公园男孩牵小狗（各测试一致，仅换加载的模型）

六、逐模型适配验证（本轮实跑）

6.1 视频 VAE：`LTX23_video_vae_bf16`

- 加载：通用 `VAELoader`；解码：`LTXVTiledVAEDecode`。产物 `adapt_gen12_videovae_00002_.mp4`（786 KB），全细节电影级（图 1）。成片标准路径。

6.2 快速预览 VAE：`taeltx2_3`

- 加载：`VAELoader`（23 MB，近零显存）；解码：`VAEDecode` 解同一批 latent。产物 443 KB，画质明显偏软（图 2）。预览专用，成片仍用视频 VAE。

6.3 音频 VAE：`LTX23_audio_vae_bf16`

- **适配发现**：专用 `LTXVAudioVAELoader` 只从 `models/checkpoints` 取，选不到 `models/vae` 的独立音频 VAE；须用通用 `VAELoader` 加载后接 `LTXVAudioVAEEncode/Decode`。
- 本轮：`EmptyAudio(3 s, 44.1 kHz, 立体声)` → 编码 → 解码 → `SaveAudio`，成功产出 `adapt_audiovae_rt_00002.flac`（45 KB），耗时 29.3 s。

6.4 fp8 文本编码器：`gemma_3_12B_it_fp8_e4m3fn`（本次新部署）

- 部署：8 路并行下载 13.2 GB（hf-mirror），safetensors 头校验 OK，部署到 `text_encoders/`。
- 加载：作为 `LTXAVTextEncoderLoader` 的 `text_encoder` 槽直接替换全精度版（无需改结构）。ComfyUI 的 `DualCLIPLoader`/`CLIPLoader`/`LTXVGemmaCLIPModelLoader` 选择器也均列出该文件。
- 本轮：出片成功 `adapt_gemmafp8_videovae_00001_.mp4`（713 KB），画质与全精度版等价（图 5，与图 1 同为基座清晰输出）。
- 价值：磁盘 13.2 GB vs 全精度 22.7 GB（小 42%），是省加载/省显存的等效替换。
- 说明：本轮为热切换（全精度版此前已常驻显存），故显存增量测得偏大（约 25 GB，含两版编码器同时驻留）；干净的显存节省量需在独立进程中单测，列为后续项。

6.5 文本投影：`ltx-2.3_text_projection_bf16`

- 状态：存在、safetensors 有效、被 ComfyUI 识别（出现在 `CLIPLoader`/`DualCLIPLoader`/`LTXVGemmaCLIPModelLoader` 的选择器中）。
- 定位：它是**合并 checkpoint 内已打包文本投影的独立/备用版**。生产 T2V 路径经 `LTXAVTextEncoderLoader` 从合并 checkpoint 读取投影（即隐式使用等价投影），本轮各次出片均依赖此投影正确工作（否则会触发 §4.1 的 3840/4096 维度报错）。独立文件作为 DualCLIPLoader 路径的备选。

6.6 Singularity LoRA：`Singularity-LTX-2.3_OmniCine_V1`

- 加载：`LoraLoaderModelOnly`（强度 1.0），显存增量约 1.3 GB。产物 `adapt_gen3_lora_videovae_00002_.mp4`（392 KB），耗时 83.7 s。
- 效果：构图重构为**电影级人物特写、暖调、无基座烧录字幕**（图 3），与作者 README（电影化镜头语言/角色一致性/消除烧录字幕）一致。偏 I2V/首尾帧/参考图，T2V 亦生效。

6.7 Licon-MSR LoRA：`LTX-2.3-Licon-MSR-V1`（本次新部署）

- 部署：ModelScope 下载 624 MB，头校验 OK，部署到 `loras/`。

- 加载：`LoraLoaderModelOnly`（强度 1.0）。产物 `adapt_licon_videovae_00002_.mp4`（778 KB），本轮冷缓存耗时 165 s、显存增量约 4 GB。
- 效果：画面在本批中**最干净锐利、构图完整、无烧录字幕**（图 4）。
- 说明：Licon-MSR 是"多主体参考（Multiple Subject Reference）"LoRA，设计用途是 **2–5 张参考图 → 视频**，完整能力需配套 **`ComfyUI-Licon-MSR` 插件**（多参考图注入）。本次以文生视频（T2V）验证其在本机可加载、可正确出片、不报错；参考图驱动的完整 MSR 能力需装插件后另测（后续项）。

6.8 五路输出对比（同提示词、同种子 101、768×512）

图1 视频VAE（成片·全细节）

图2 tae（预览·偏软）

图3 Singularity LoRA（电影级特写）

图4 Licon-MSR LoRA（构图干净完整）

图5 gemma-fp8（与全精度等价·基座输出）

> 图 1/2：同一批 latent 分别经视频 VAE 与 tae 解码——直观体现两档 VAE 画质差异（tae 明显偏软）。图 1/5：全精度 gemma 与 fp8 gemma 输出质量等价（均为基座清晰输出，含基座偶发烧录字幕）。图 3/4：两个 LoRA 分别把画面重构为电影级特写、干净完整构图，且均消除烧录字幕。

七、适配数据汇总（本轮重跑）

测试	覆盖模型	状态	墙钟	执行	显存增量	产物
基线双解码	视频VAE + tae + gemma(全精度) + 文本投影	success	48.0 s	46.5 s	≈0	786 KB + 443 KB mp4
Singularity LoRA	Singularity + 视频VAE	success	84.2 s	83.7 s	+1.3 GB	392 KB mp4
Licon-MSR LoRA	Licon-MSR + 视频VAE	success	168.1 s	165.4 s	+4.0 GB	778 KB mp4
音频往返	音频VAE	success	30.0 s	29.3 s	波动	45 KB flac

gemma-fp8 出片	gemma-fp8 + 视频VAE + 文本投影	success	120.2 s	119.9 s	+25.2 GB（热切换）	713 KB mp4

- 采样吞吐：768×512 下约 1.9 s/步，20 步约 40 s；
- 热启动单次出片约 48–84 s（LoRA 冷缓存首跑更久，Licon 冷缓存 165 s）；gemma-fp8 因需加载 13.2 GB 新编码器进显存，本次 120 s；
- 五组测试全部成功，`node_errors` 为空，无 dtype/算子/内核报错。

八、关键适配发现与建议

1. **独立 VAE 一律用通用 `VAELoader`**（读 `models/vae`）：视频/音频/tae 均适用。专用 `LTXVAudioVAELoader` 只读 `models/checkpoints`，不能加载独立音频 VAE——最易踩的坑。
2. **合并 checkpoint 是前提**：拆分 transformer 缺文本投影会导致 3840/4096 维度不匹配报错。
3. **gemma fp8 vs 全精度可互换**：fp8_e4m3fn 直接替换 `LTXAVTextEncoderLoader` 的 text_encoder 槽即可，出片等价，磁盘小 42%，适合显存/加载敏感场景。
4. **tae 仅预览**：成片必须用视频 VAE。
5. Singularity LoRA 偏 I2V/首尾帧/参考图，能抑制基座烧录字幕；追求最佳效果建议补作者推荐的蒸馏基座 + 官方 i2v 工作流。
6. Licon-MSR LoRA 需 `ComfyUI-Licon-MSR` 插件才能发挥多主体参考（2–5 参考图）完整能力；纯 T2V 已验证可加载可出片。
7. **基座字幕烧录**：dev-fp8 基座即便负向提示含"no subtitles"仍偶发烧录乱码；挂 LoRA（Singularity/Licon）可显著缓解。
8. **下载加速**：无 aria2c 时可用 curl range 8 路并行（各段断点续传）绕过 hf-mirror 单连接限速，实测 4→17 MB/s。

九、结论

ComfyUI + LTX-Video 2.3（22B fp8）在 AMD Instinct MI308X / ROCm 7.2 上**部署完成、跑通稳定、可复现、可经 API 批量生成。全部 7 个交付模型**（视频/音频/快速预览 VAE、fp8 文本编码器、文本投影、Singularity 与 Licon-MSR 两个 LoRA）均已部署并实跑验证通过：加载无误、前向无算子/内核错误、产物正常。其中 gemma-fp8 与 Licon-MSR 为本次新部署并验证。唯一需注意的适配点是独立音频 VAE 须用通用 `VAELoader` 加载。整体管线满足生产化文生视频的出片与迭代需求。

十、后续可选项

1. **Licon-MSR 完整能力**：装 `ComfyUI-Licon-MSR` 插件，用 2–5 张参考图跑多主体参考视频（其设计主场景）；
2. **gemma-fp8 净显存收益**：在独立进程中单测 fp8 vs 全精度的显存峰值，给出干净的省显存量；
3. **加音频出片**：接 LTX 的 AV 节点链（`LTXVEmptyLatentAudio`/`LTXVConcatAVLatent`/`LTXVAudioVAEDecode`）做带声音的联合音视频；
4. **图生视频/首尾帧**：`LTXVImgToVideo` 系列 + Singularity 首尾帧控制；
5. **提质提速**：接空间超分 ×2 放大；挂蒸馏 LoRA + 少步采样（8 步）。

附录 A：可复现步骤（服务器 10.1.29.48，容器 `comfyui`）

A.1 补齐两个模型

```
# Licon-MSR LoRA (ModelScope)
BASE_L=https://modelscope.cn/models/LiconStudio/LTX-2.3-Multiple-Subject-Reference/resolve/master
curl -L -C - "$BASE_L/LTX-2.3-Licon-MSR-V1.safetensors" -o /srv2/comfyui/ComfyUI/models/loras/LTX-2.3-Licon-MSR-V1.safetensors

# gemma-fp8 (HuggingFace, 仅此源; 8 路并行加速见附录 B)
BASE_G=https://hf-mirror.com/GitMylo/LTX-2-comfy_gemma_fp8_e4m3fn/resolve/main
# 单线程: curl -L -C - "$BASE_G/gemma_3_12B_it_fp8_e4m3fn.safetensors" -o .../text_encoders/gemma_3_12B_it_fp8_e4m3fn.safetensors
```

A.2 各模型实跑关键片段（经 ComfyUI API）

```
# 视频 VAE 解码
"20":{"class_type":"VAELoader","inputs":{"vae_name":"LTX23_video_vae_bf16.safetensors"}}
"12":{"class_type":"LTXVTiledVAEDecode","inputs":{"vae":["20",0],"latents":["11",0],"horizontal_tiles":2,"vertical_tiles":2,"overlap":6,"last_frame_fix":False}}
# tae 预览
"21":{"class_type":"VAELoader","inputs":{"vae_name":"taeltx2_3.safetensors"}}
"22":{"class_type":"VAEDecode","inputs":{"samples":["11",0],"vae":["21",0]}}
# gemma-fp8: 替换文本编码器槽
"2":{"class_type":"LTXAVTextEncoderLoader","inputs":{"text_encoder":"gemma_3_12B_it_fp8_e4m3fn.safetensors","ckpt_name":"ltx-2.3-22b-dev-fp8.safetensors","device":"default"}}
# Singularity / Licon-MSR LoRA
"30":{"class_type":"LoraLoaderModelOnly","inputs":{"model":["1",0],"lora_name":"Singularity-LTX-2.3_OmniCine_V1.safetensors","strength_model":1.0}}
"30":{"class_type":"LoraLoaderModelOnly","inputs":{"model":["1",0],"lora_name":"LTX-2.3-Licon-MSR-V1.safetensors","strength_model":1.0}}
# 音频 VAE 往返(用通用 VAELoader)
"a1":{"class_type":"EmptyAudio","inputs":{"duration":3.0,"sample_rate":44100,"channels":2}}
"a2":{"class_type":"VAELoader","inputs":{"vae_name":"LTX23_audio_vae_bf16.safetensors"}}
"a3":{"class_type":"LTXVAudioVAEEncode","inputs":{"audio":["a1",0],"audio_vae":["a2",0]}}
"a4":{"class_type":"LTXVAudioVAEDecode","inputs":{"samples":["a3",0],"audio_vae":["a2",0]}}
"a5":{"class_type":"SaveAudio","inputs":{"audio":["a4",0],"filename_prefix":"adapt_audiovae_rt"}}
```

附录 B：gemma-fp8 多线程加速下载（无 aria2c 时）

```
URL=https://hf-mirror.com/GitMylo/LTX-2-comfy_gemma_fp8_e4m3fn/resolve/main/gemma_3_12B_it_fp8_e4m3fn.safetensors
SZ=$(curl -sIL "$URL" | grep -i '^content-length:' | tail -1 | awk '{print $2}' | tr -d '\r')
PARTS=8; chunk=$(( (SZ+PARTS-1)/PARTS ))
for i in $(seq 0 $((PARTS-1))); do
  s=$((i*chunk)); e=$((s+chunk-1)); [ $e -ge $SZ ] && e=$((SZ-1))
  curl -sL -C - -r ${s}-${e} "$URL" -o part_$i &
done; wait
cat part_0 part_1 part_2 part_3 part_4 part_5 part_6 part_7 > gemma_3_12B_it_fp8_e4m3fn.safetensors
# 实测把 hf-mirror 从单线程 ~4 MB/s 提速到聚合 ~17 MB/s
```

附录 C：产物与留档

文件	说明

output/adapt_gen12_videovae_00002_.mp4	视频 VAE 成片（786 KB）
output/adapt_gen12_tae_00002_.mp4	tae 预览片（443 KB）
output/adapt_gen3_lora_videovae_00002_.mp4	Singularity LoRA 成片（392 KB）
output/adapt_licon_videovae_00002_.mp4	Licon-MSR LoRA 成片（778 KB）
output/adapt_gemmafp8_videovae_00001_.mp4	gemma-fp8 成片（713 KB）
output/adapt_audiovae_rt_00002.flac	音频 VAE 往返产物（45 KB）
/srv2/comfyui/ref/*	Singularity 与 Licon 的作者 README、官方工作流

本报告全部数据均来自服务器实测，工作流、命令与产物完整留档，可独立复现。