I’ve Got 99 Problems, but LLMs Ain’t One!

Cedric Clyburn · Senior Developer Advocate, Red Hat

Ai4 Las Vegas · 2025

Ai4

Why private and local AI?

83% of organizations are moving workloads back to private cloud or on-prem for privacy and cost (Barclays CIO Survey 2024).

I’ve Got 99 Problems, but LLMs Ain’t One!

Cedric Clyburn · Senior Developer Advocate, Red Hat

Ai4

Why private and local AI?

83% of organizations are moving workloads back to private cloud or on-prem for privacy and cost (Barclays CIO Survey 2024).

Control & Privacy
Keep data on your infra, control updates and versions

Cost Predictability
Avoid unpredictable egress/inference bills

Customization
Tune models, add guardrails, ship your cadence

Portability
Run on Linux, Kubernetes, or bare metal

Let's get started!

About me

Cedric Clyburn

Senior Developer Advocate, Red Hat

Organizer, KCD New York

Open source + community builder · @cedricclyburn

Technical Educator

- RAG vs Fine-Tuning
- Ollama
- vLLM

Agenda Open Source Model Explosion

Foundation models

Open models: what's new in OSS

Llama 3.x, Qwen, DeepSeek, Mistral, Gemma

instruct base MoE

Modalities

Text, vision, audio, diffusion, multi-modal

VLM (Llava, Idefics) ASR/TTS

Diffusion

Why hosted APIs aren't always enough

Ecosystem

Tokenizers, safetensors, GGUF, serving runtimes

llama.cpp vLLM TGI GGUF

safetensors

Private & sovereign AI options

How to deploy: local → production

Tooling: llama.cpp, vLLM, TGI

Cut GPU cost by ~50%: quantization

Options to use LLMs today

Private and Sovereign AI

Own your stack from weights → runtime → platform.

Infra
Linux, GPUs, Kubernetes, observability

Serving
llama.cpp (CPU/GGUF) & wrappers, vLLM (GPU), TGI

Governance
Security, access, model registries, evaluations

Transformers

llama.cpp

vLLM

What if we cut costs in half?

Quantization shrinks weights and KV cache → fewer/lower-VRAM GPUs, faster tokens/sec.

What it is

Open-source LLM serving (UC Berkeley).

High-throughput, low-latency GPU inference with PagedAttention and efficient KV cache.

Strengths

- Excellent throughput with batching/continuous batching
- OpenAI-compatible server; easy clients
- Works with HF models; supports tensor/quant formats

Trade-offs

- Needs decent GPUs and VRAM; plan for scheduling
- Quantization can hit accuracy; mix, layer-wise sensitivity matters

Approaches

- AWQ, GPTQ, GQA-aware, INT8/INT4, FP8

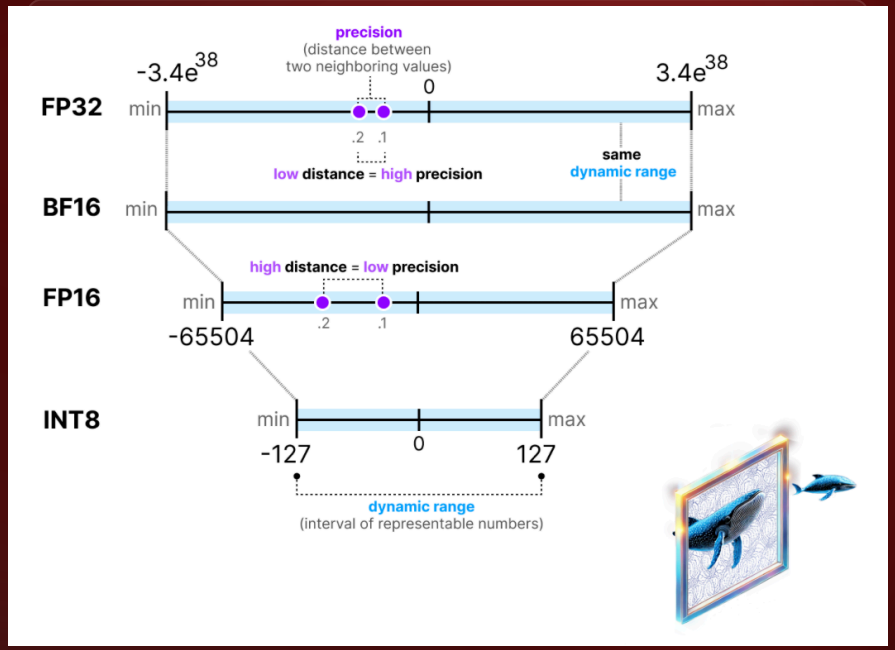
Typical wins

- 2× VRAM reduction, similar quality with right calibration

Trade-offs

- Single-node vs. multi-node (tensor parallel)

OpenAI-compatible server



Quantization demo

Local → Production

Local

- llama.cpp binary, GGUF weights
- vLLM single-GPU serve for quick iteration
- Experiment in notebooks/Transformers before serving

Production

- Kubernetes + GPU scheduling (time-slicing/MIG)
- Autoscaling, observability, circuit-breakers, canaries

Local → Production

Takeaways

Local

- llama.cpp binary, GGUF weights
- vLLM single-GPU serve for quick iteration
- Experiment in notebooks/Transformers before serving

Production

- Kubernetes + GPU scheduling (time-slicing/MIG)
- Autoscaling, observability, circuit-breakers, canaries

SRE checklist: health probes, request budgets, token limits, logging redaction, evals, red-team prompts, rollout policy.

Thanks!

Resources

- Quantization accuracy at scale (Red Hat article)
- LLM Compressor (quantization library)
- vLLM (OpenAI-compatible server)
- Red Hat AI models on Hugging Face
- ramalama (containers / local LLM tooling)
- llama.cpp (GGUF inference)

Follow

@cedricclyburn · YouTube, X/Twitter, GitHub, LinkedIn

LinkedIn

Slides

Takeaways

Open models are thriving across sizes and modalities

Private + sovereign A… …ve loca… control, privacy, and …PU/edg… governance …PU scale

Quantization can halve GPU needs with minimal quality loss

You might still have 99 problems – LLMs ain't one

Ai4