Computer-Use Demo Multi-Provider Refactor Plan

Prepared for: MCPWorld maintainers

Purpose: Establish a two-page implementation plan to add provider-agnostic LLM support.

Section 1: Strategic Objectives

1. Decouple Anthropic-specific logic from the agent sampling loop while preserving existing behaviour.
2. Support drop-in OpenAI-compatible local endpoints (e.g., Ollama, vLLM, HF TGI).
3. Maintain Streamlit UX, evaluator hooks, and tooling fidelity during the transition.

Section 2: Workstream Breakdown

Workstream A - Provider Abstraction Layer (10-12 hrs)

- Design provider-agnostic message/tool dataclasses and adapters.
- Refactor sampling_loop to dispatch via provider strategy objects.
- Implement Anthropic adapter using current beta.messages flow.
- Implement OpenAI-style adapter targeting local hosts; cover error handling.

Workstream B - Tool Schema Normalisation (6-8 hrs)

- Convert BaseAnthropicTool into provider-neutral ToolSpec interface.
- Generate provider-specific payloads at call time (Anthropic betas vs. OpenAI functions).
- Update MCP client to translate tool metadata into the unified schema.
- Verify backward compatibility with existing tool versions (20241022, 20250124).

Workstream C - UX and Config Integration (4-5 hrs)

- Extend Streamlit sidebar to manage provider selection, local endpoint URL, auth tokens.
- Persist provider-specific settings in ~/.computeruse (avoid Anthropic-only paths).
- Update headless scripts to accept provider flag and propagate through CLI args.

Workstream D - Validation and QA (5-6 hrs)

- Build unit tests for provider adapters, tool translation, and MCP bridge.
- Add integration harness running against mocked Anthropic responses and local OpenAI server.
- Perform manual regression on Streamlit flow with both providers enabled.
- Document fallback plan if local endpoint fails (graceful error surfacing).

Section 3: Dependencies and Risks

Dependencies

- Access to local LLM runtime (Ollama or vLLM) with OpenAI-compatible API.
- Test credentials for Anthropic to run regression suite.

Risks and Mitigations

- Schema drift between providers => maintain adapter conformance tests.
- Tool action parity issues => add contract tests for keyboard and mouse actions.
- Performance regressions with local models => benchmark key tasks post-integration.

Section 4: Timeline and Milestones

Week 1: Complete Workstream A and scaffold Workstream B (Milestone: dual-provider sampling loop)

Week 2: Finish Workstreams B and C (Milestone: Streamlit plus headless flows toggle providers).

Week 3: Execute Workstream D, fix regressions, prepare release notes (Milestone: QA sign-off).