

Progression

ChatGPT Recommended Changes to original progression

1. **Contracts still first — keep exactly as is.**
Everything hangs off versioned schemas + transport.
2. **Move “AI Bridge v0” earlier (right after contracts).**
Getting tick alignment, seq-nums, backpressure, and heartbeats stable early will de-risk everything else. You can exercise it with dummy obs/acts while the game bits are coming online.
Exit stays the same: obs → trivial policy → act loopback under latency budget.
3. **Build a thin vertical slice before splitting obs vs actions.**
Instead of “Observation v0” then “Action v0”, do **Observation v0 + Action v0 together** to close the loop once. You’ll spot cadence/GC hitches and anticheat spikes sooner.
4. **Pull the Evaluation Harness up (before the baseline policy).**
You want repeatable scenarios in place so every later step gives you numbers, not vibes.
5. **Then add the Baseline scripted policy + logging** and proceed with the rest as you listed.

New Suggested order

1. **Contracts (schemas + transport)**
2. **AI Bridge v0 (local loopback, tick/seq/heartbeats/backpressure)**
3. **Client-side Controller (toggle AI↔human + safe idle)**
4. **Vertical Slice v0: Observation v0 + Action v0** (small, steady cadence; token-bucket)
5. **Evaluation Harness + deterministic scenarios** (one-click report)
6. **Baseline scripted policy + full logging** (latency & stability baseline)
7. **Learning & Inference v1** (mock env → live swap; beat baseline)

8. **Curriculum + rollout collection** (nav → survival → simple combat/crafting)
9. *Action Interface v2 (macros/Baritone/A/JPS + safety)**
10. **Observation v2** (entities/affordances/compact voxel; compression if needed)
11. **AI Bridge v2** (versioning/migrations, remote deploy, resilience)
12. **Headless client path (optional, in parallel once eval exists)**