Disconnected Is the Baseline: Designing Agentic AI for the Tactical Edge

By: Nick Weir PhD, VP of Mission Engineering

Agentic AI at the tactical edge is not a cloud product with a field kit. It is a different class of system.

In garrison, an AI system can assume stable networks, abundant compute, managed identity, cloud-hosted models, and help desk access. At the edge, those assumptions fail first. Connectivity is denied, degraded, intermittent, or limited ("DDIL"). Power is constrained. Operators are under time pressure. Data is incomplete. **The system may have to support decisions while fully isolated from the outside world.**

That changes the architecture. The issue is not whether an AI tool can run near the mission, but instead whether the system can continue to produce useful, governed, and trusted outcomes when reach-back disappears.

Edge AI must be disconnected-first. It must run locally, degrade gracefully, synchronize opportunistically, and preserve human authority where decisions carry tactical, legal, or escalation consequence. The battlefield evidence from Ukraine points in the same direction: the most mature uses of AI-enabled autonomy are narrow, mission-specific functions such as target recognition, navigation, tracking, and decision support, not generic frontier agents running every mission from a prompt box. We're already seeing this in the real world: Ukraine has made significant progress in partial autonomy while human oversight remains critical for engagement decisions (as detailed by CSIS), and Ukrainian teams are training smaller AI models on focused datasets for onboard processing on limited chips.

DDIL Is Not an Exception Case

A system that depends on reach-back isn't really an edge system.

Most AI architectures, even those purportedly built for the tactical edge, still treat connectivity as a background condition. Models live in the cloud. Data retrieval depends on centralized stores. Updates, logging, monitoring, and authentication assume reliable network paths. That may be acceptable for many headquarters workflows, but it is not acceptable for contested environments.

In DDIL conditions, the design center has to invert. **The local node must be able to operate with no external dependency for the core mission workflow.** That means local models, local caches, local policies, local audit capture, local identity assumptions, and local fallbacks. When bandwidth returns, the system should synchronize selectively. It should not wait for the network to become useful.

The Services have recognized this. For example, the Army's Unified Network Plan 2.0 published in early 2025 explicitly calls for hybrid compute in support of tactical formations in DDIL environments, integrated with Army cloud strategy. The same plan emphasizes moving timely intelligence where it is needed for local processing, reducing undue complexity while maintaining governance.

Leaders should evaluate edge AI systems by their failure modes. What happens when SATCOM drops? What happens when the model registry is unreachable? What happens when two nodes can see each other, but neither can reach the cloud? What data stays local? What synchronizes later? What authority does the system retain? In a peer conflict, all of these scenarios can and will happen.

Small Models and Narrow Planners Win at the Tactical Edge

At the edge, the best model often isn't the "frontier", it's the one that fits the mission.

The reflexive instinct in AI procurement is to chase the largest, most recent available model from a hyperscaler model provider. That instinct is understandable, but it is also often wrong at the tactical edge.

A frontier model can be valuable when connectivity, classification, latency, and cost allow it, but tactical systems cannot assume that luxury. Edge systems are still subject to the same Size, Weight, and Power (SWaP) and networking constraints as before, and adding AI to the mix didn't somehow eliminate those constraints. **Optimizing a cloud system for low bandwidth, high latency, intermittent connections to the tactical edge doesn't cut it.** A small vision model that reliably identifies a known priority vehicle class from mission-relevant drone imagery matters more than a generalist frontier model that can write text summaries of 30 image types in 100 languages when connected to the cloud.

Importantly, these **purpose-built systems leveraging smaller models build user trust**. They are more predictable and more explainable. Importantly, smaller models built or tuned for specific mission uses fit into well-governed, structured agentic workflows in much the same way that a frontier generalist model would.

The paradigm of using smaller models purpose-built for specific missions may not last forever. Each time a new set of small, open-weight models are released, they outstrip performance of behemoth models that were the frontier two or three model generations ago. At some point it

may become trivial to deploy generalist models on low-SWaP compute that's realistic to deploy and operationalize at the tactical edge. Decision-makers can assess generalist models against purpose-built systems through consistent evaluation of models in operationally relevant workflows as covered in an earlier Command Paper on AI agent evaluation.

Senior leaders should require evidence that vendors have mapped each workflow to the right model. Ask where the system uses local models, where it escalates to remote models, what happens when escalation is unavailable, and how model performance is validated against mission-specific data.

A model that cannot fit the edge is not a capability. It is a dependency.

Sustainment Is Part of the System

Training, batteries, spares, heat, updates, and data labels are not support tasks. They are mission enablers.

Many edge AI efforts fail because the architecture is treated as software while the deployment behaves like a weapon system. The model is only one part of the capability. The operator interface, power budget, thermal profile, spare parts, local data practices, update path, rollback plan, and user training determine whether the system survives contact with the mission. Fortunately, personnel acquiring tactical edge AI systems know this and are asking the right questions. The industry needs to catch up by bringing systems that deliver the right answers.

User training is worth extra emphasis. If a warfighter only touches the system during a last-minute deployment refresher, the system will be foreign when it matters. Whenever possible, the operator experience between deployments should match the deployed experience. The tool used at the desk should resemble the tool used in the field.

Leaders should fund edge sustainment as a warfighting function. That means AI enablement in the schoolhouses, operator confidence measures, realistic DDIL exercises, and spares. **Training and logistics are not secondary to tactical AI; they are part of whether the capability becomes operationally real.**

Centurion by Legion Intelligence is built on exactly this principle: an integrated edge system, not a loose bundle of components. Centurion is designed to run disconnected, and Centurion nodes can integrate across a mesh network to balance workloads and support model fallbacks when networking is available. Systems architected this way are designed to operate where internet access is unavailable, bandwidth is constrained, or transmitting is operationally risky.

The Edge Is a Tempo Problem

The tactical edge does not need more information. It needs better decisions faster.

AI systems at the edge must reduce cognitive burden. They should not flood operators with every possible detail. They should surface what changed, why it matters, what the system did, what it recommends, and where human judgment is required.

That is the core tempo problem. The Observe-Orient-Decide-Act (OODA) loop is tighter at the edge because the environment changes faster, the adversary interferes more directly, and the cost of delay is higher (particularly as they too are increasingly AI-enabled). AI should compress search, triage, correlation, and routine execution so humans can spend scarce attention on command judgment. Anyone who has sighed when faced with a scrolling wall of response text coming from an AI model can appreciate the challenge this would present in a truly life-and-death scenario.

The credible architecture is a resilient system of local execution, narrow autonomy, tiered model fallback, governed workflows, mesh-aware coordination, and human authority. It should run when isolated. It should improve when connected. It should synchronize without becoming dependent. It should reduce operator cognitive burden without hiding uncertainty or seizing decisions that belong to the user.

The measure of the system is not how much it can automate, but how reliably it preserves human judgment under pressure.

Build for Isolation, Not Convenience

Edge AI will not earn trust because it performs well in a lab, connects cleanly to a cloud service, or demonstrates impressive reasoning in a controlled environment. It will earn trust when it continues to support the operator after the network fails, the data is incomplete, the hardware is constrained, and the mission still has to move.

That requires a different design philosophy. Disconnected operation cannot be an afterthought. Local models, governed workflows, tiered autonomy, selective synchronization, operator training, and sustainment have to be built into the system from the start. What holds up at the edge is resilience, not elegance. The industry at large needs to take this to heart before we can realistically expect holistic adoption of agentic AI systems in the battlefield.

Leaders should demand evidence that an AI system can operate in isolation, degrade gracefully, preserve human authority, and reduce cognitive burden under real mission conditions. Anything less is a connected system waiting for the environment to cooperate.

At the edge, resilience is the foundation of trust.

Continue the series

Follow Legion Intelligence on LinkedIn: linkedin.com/company/legion-intel

Explore all Command Papers: legionintel.com/command-papers

New papers published every 1–2 weeks.