



[Home](#) [Licensing](#) [Patents](#) [Articles](#)

## Underwater Robotic Operations Without Connectivity

by [Nick Clark](#) | Published March 27, 2026 | [PDF](#)

Radio waves do not propagate through seawater. Underwater robots communicate through acoustic links with bandwidth measured in kilobits per second and latency measured in seconds per kilometer. Memory-resident execution enables autonomous underwater vehicles to carry their complete mission as a persistent, self-evaluating execution context that adapts to observed conditions, makes governed decisions about inspection priorities, and continues meaningful operations through extended periods without any surface connectivity.

---

### The communication barrier underwater

Underwater communication is one of the most challenging environments in engineering. Radio frequencies that enable megabit communication in air attenuate within meters of seawater. Acoustic modems provide communication at ranges of several kilometers but at bandwidths of single-digit kilobits

per second with multi-second propagation delays. Optical communication works at short range in clear water but fails in turbid conditions.

The practical consequence is that autonomous underwater vehicles (AUVs) operating at depth are fundamentally disconnected from surface operators. A subsea infrastructure inspection AUV at several hundred meters depth on a multi-hour mission has, at best, intermittent acoustic contact with its support vessel. The contact provides enough bandwidth for status telemetry but not enough for real-time guidance or mission replanning.

Current AUVs handle this through pre-programmed survey patterns. The vehicle follows a lawnmower path over the survey area, collects data at predetermined intervals, and surfaces for data offload and mission update. This approach works for uniform survey tasks but fails for inspection missions where the AUV needs to deviate from the pre-planned path to investigate anomalies detected during the survey.

## Why pre-planned missions waste operational time

Pre-planned survey patterns collect data uniformly regardless of what the data reveals. An AUV inspecting a subsea pipeline that detects a potential anomaly cannot pause to investigate it further unless the pre-planned mission included contingency behaviors for that exact scenario. Instead, the anomaly is flagged in the collected data, the vehicle completes its pre-planned path, surfaces, offloads data, and a human operator reviews the data and programs a follow-up mission to investigate the anomaly.

This cycle of survey, surface, review, and re-deploy wastes hours or days of operational time and vessel cost for each iteration. In offshore energy operations where vessel day-rates exceed six figures, these delays have direct financial impact. In defense applications where underwater survey results are time-sensitive, the delays have operational impact.

## How memory-resident execution addresses this

Memory-resident execution enables the AUV to carry its mission as a persistent semantic object that self-evaluates and mutates based on observed conditions. When the AUV detects a potential anomaly, the execution object evaluates the observation against its mission objectives, proposes a plan mutation to investigate the anomaly, validates the mutation against its governance policy including time constraints, battery reserves, and safety margins, and executes the investigation if the mutation is approved.

The AUV does not need to surface for human approval. The governance that constrains the investigation, how much time to spend, how close to approach, what safety margins to maintain, is embedded in the execution object. The decision is governed, auditable, and fully recorded in lineage.

For multi-day missions, the execution object accumulates observations across the entire mission, identifying patterns that would only be visible to a human analyst reviewing the complete dataset after retrieval. The AUV can refine its survey strategy as it collects data, focusing attention on areas of interest and reducing time spent on areas that initial passes reveal to be unremarkable.

## What implementation looks like

An AUV deploying memory-resident execution carries its mission object on onboard compute alongside its navigation, sensor processing, and control systems. The mission object defines the inspection objectives, survey constraints, anomaly detection criteria, and investigation governance. The AUV executes the mission object's self-evaluation cycle continuously throughout the dive.

For offshore energy operators, this reduces the number of dive cycles needed to complete an inspection from multiple pre-planned dives to a single adaptive dive. The AUV investigates anomalies during the survey rather than flagging them for follow-up. The operational cost reduction is proportional to the number of avoided re-deployment cycles.

For ocean science, memory-resident execution enables AUVs to conduct adaptive sampling missions where the sampling strategy evolves based on what the instruments detect. An oceanographic AUV that detects an unexpected temperature gradient can investigate the gradient's extent and structure during the same dive rather than completing a pre-planned grid and returning for a targeted follow-up.

For defense, underwater survey missions gain the ability to adapt to discovered conditions without surfacing to communicate with operators, maintaining operational security while increasing the quality and completeness of the survey data collected per mission.

[Memory-Resident Execution All 21 steps →](#)

Persistent objects that execute without orchestration.

Patent

[US 19/538,221](#) · filed

Primary Technical Disclosure

◦ [Memory-Resident Execution: Persistent Semantic Objects Without Orchestration](#)

Secondary Technical

◦ [Six-Action Execution Evaluation Cycle: Parse, Evaluate, Select at Every Node](#)

◦ [Cognition-Authority-Execution Separation: Reasoning Cannot](#)

[Authorize Action](#)

◦ [Dormancy as First-Class Execution State: Valid Suspension Without Failure](#)

◦ [Semantic Backoff: Retry Pacing From Execution](#)

[Outcomes Rather Than Fixed Timers](#)

◦ [Wake Triggers for Dormancy Exit: Explicit Reentry Conditions in Memory](#)

◦ [Persistent Polling Behavior: Autonomous Condition Evaluation Without Schedulers](#)

◦ [Intent Refinement During Execution: Adaptive Objectives Without Re-Instantiation](#)

◦ [Compositional Execution Through Recursive Delegation: Parent-Child Lineage Tracking](#)

◦ [Negative Capability Signals: Recording What Cannot Be Done](#)

[as Structured Constraint](#)

◦ [Swarm-Based Execution Emergence: Coordinated Behavior Without Centralized Control](#)

◦ [Latency and Failure as Semantic Signals: Structured Inputs From Adverse Conditions](#)

◦ [LLM as Advisory Execution Node: Inference Without Authority Over Agent State](#)

◦ [Append-Only Memory Field: Complete Execution Lineage Through Immutable Records](#)

Applications (General)

◦ [Serverless Execution Without Cold Starts or State Loss](#)

◦ [Long-Running Autonomous Workflows Without External Orchestration](#)

◦ [Drone Operations Surviving Disconnection](#)

◦ [Deep Space Agent Execution Without Ground Control](#)

◦ [Underwater Robotic Operations Without Connectivity](#)

◦ [Rural Healthcare Agents Surviving Intermittent Connectivity](#)

◦ [Operations in Infrastructure-Destroyed Environments](#)

◦ [Offline Financial Transaction Agents](#)

Applications (Specific)

[◦ Cloudflare Durable Objects Made State Local. The Objects Still Need Orchestration.](#)[◦ Azure Service Fabric Actors Are Addressable. They Are Not Autonomous.](#)[◦ Akka Perfected the Actor Model. Actors Still React Instead of Self-Execute.](#)[◦ Orleans Made Virtual Actors Practical. The Actors Still Execute on Request.](#)[◦ Dapr Provides a Sidecar Runtime for Microservices. The Services Still Need External Orchestration.](#)[◦ wasmCloud Runs WebAssembly Actors. The Actors Wait for Messages.](#)[◦ Spin Made WebAssembly Serverless. The Functions Are Still Trigger-Based.](#)[◦ Fermion Built the WebAssembly Cloud. The Cloud Hosts Functions, Not Self-Executing Objects.](#)[◦ Fly Machines Made Micro-VMs Fast. The VMs Still Need External Orchestration.](#)[◦ Railway Simplified Application Deployment. The Applications Still Depend on External Execution Triggers.](#)  
[Memory-Resident Execution overview →](#)

AQ  
deterministic  
autonomy

Legal

Subject to one or more pending U.S. and international patent applications, see [Patents](#) for the current list and status. No license, express or implied, is granted. Any use requires a separate written agreement—see [Licensing](#). Patent applications referenced on this site are pending. Claim scope, if any, is subject to examination and may issue in altered form or not at all. See [Legal](#) for terms and conditions.

Adaptive Query™ is a trademark of Nicholas Clark. U.S. federal registration is pending. federal registration. AQ™, AQ Inside™, Adaptive Index™, Adaptive Network™, Semantic Agent™, @AQ™, AQID™, and Adaptive Coin™ are used as trademarks in connection with the Adaptive Query platform and brand. Other names may be trademarks of their respective owners.

Platform operated by Adaptive Query LLC, which provides patent and trademark licensing services. Copyright © 2025-2026 Nicholas Clark. All rights reserved.

Last updated: 2026-03-03



- [Inventive Steps](#)
- [Licensing](#)
- [Patents](#)
- [Articles](#)
- [Legal](#)
- [Opportunities](#)
- [Sitemap](#)



- 
- [nick@qu3ry.net](mailto:nick@qu3ry.net)
- 72 28 14 36 01



[Invented by Nick Clark](#) | Founding Investors: Devin Wilkie