SYM-AM-26-073

EXCERPT FROM THE PROCEEDINGS

OF THE

TWENTY-THIRD ANNUAL
ACQUISITION RESEARCH SYMPOSIUM AND
INNOVATION SUMMIT

WEDNESDAY, MAY 6, 2026 SESSIONS
VOLUME I
“ACCELERATING WARFIGHTING CAPABILITIES”

Embedded Capability Development: A Case Study in Rapid Missile Prototyping and Transition

Published: April 30, 2026

The research presented in this report was supported by the Acquisition Research Program, Graduate School of Defense Management at the Naval Postgraduate School.

Embedded Capability Development: A Case Study in Rapid Missile Prototyping and Transition

Maj Dillon T. Pierce, PhD—is an active-duty Marine Corps officer serving as Chief Engineer for the Marine Corps Warfighting Laboratory Rapid Capabilities Office and as a Research Assistant Professor in the Space Systems Academic Group at the Naval Postgraduate School. He completed his doctoral research through the Marine Corps PhD Program–Technical (PHDP-T), where his dissertation focused on agile, low-cost missile development and flight validation. His current work centers on rapid capability development, missile prototyping, and transition-oriented defense innovation, which provides the basis for this paper.

Mr. Christopher Reasonover—is a government civilian detailed from Naval Surface Warfare Center Dahlgren Division and currently serving as the Precision Fire Control Program Officer within Code 352 (Kinetic Weapons) in the Office of Naval Research. His work spans the development, technical oversight, and transition of precision fires technologies relevant to future naval warfighting capability. He has an extensive background in radar and missile development, informed by 25 years of engineering experience at Dahlgren supporting advanced weapons and fire-control efforts. He holds a BS in Aerospace Engineering from Auburn University.

Abstract

This paper presents embedded capability development as a practical execution approach for integrating requirements refinement, technical maturation, and transition preparation within a single collaborative government-industry development effort. In this model, a uniformed Service member is embedded within the technical team and serves both as a hands-on developer and as the principal government lead for the day-to-day refinement of evolving operational and system-level requirements. The approach addresses two persistent problems in defense capability development: promising science and technology efforts that fail to mature into transition-ready capabilities, and formal development processes that separate requirements, development, resourcing, and transition into sequential activities that delay delivery to the warfighter. The paper examines the model through an ongoing case study in low-cost tactical missile development and describes the organizational structure, collaborative industry engagement, and enabling tools used to implement it, including a government reference architecture and government reference design. The case suggests that embedded capability development can provide a practical complement to recent strategic-level reform guidance by showing how a program can organize itself to integrate technical learning, evolving requirements, and transition preparation within one collaborative development effort.

Introduction

This paper examines an embedded capability development approach in which a uniformed Service member operates directly within a collaborative government-industry technical development effort while serving both as a hands-on developer and as the principal government lead for the day-to-day refinement of evolving operational and system-level requirements. Rather than treating requirements, development, and transition as sequential activities, the approach integrates them within a single iterative effort so that design decisions, prototype maturation, and requirement refinement inform one another continuously.

The approach is intended to address two persistent challenges in defense capability development. First, promising science and technology (S&T) efforts often fail to mature into fielded capability because the broader conditions required for transition are not established alongside technical progress. Second, formal capability development processes have historically struggled to adapt once technical and operational realities begin to evolve, particularly when requirements, resourcing, and acquisition activity remain too disconnected from ongoing technical learning. Embedded capability development is presented here as one practical execution model for efforts that require meaningful technical development, involve transition complexity, and benefit from close collaboration across government and industry.

The paper first reviews the limitations of current capability development approaches, then introduces the model, and finally examines its implementation in an ongoing low-cost tactical missile development effort.

Background: Limitations of Current Capability Development Approaches

Current military capability development approaches continue to exhibit two recurring failure patterns: promising technical efforts that do not transition beyond prototype demonstration, and formal development pathways that struggle to deliver integrated capability at the speed demanded by modern conflict.

The S&T Valley of Death

A longstanding observation in defense innovation is the repeated failure of promising capabilities to cross the “Valley of Death,” the gap between technology development and acquisition. As the Government Accountability Office (GAO) and other Department of Defense (DoD) observers have noted, this gap is driven in part by a mismatch in maturation expectations: acquisition organizations typically require a higher degree of technical and programmatic maturity than S&T organizations are funded or incentivized to provide (DoD, 2007; GAO, 2015). The problem is compounded by the fact that successful transition depends not only on technical performance, but also on requirements alignment, integration planning, resourcing, sustainment, and a viable acquisition pathway. Because these functions are often distributed across separate organizations, transition frequently breaks down even when the underlying technology shows promise (Murray, 2023). In practice, promising prototypes may demonstrate feasibility, and even support limited operational employment, while still failing to establish the broader conditions necessary for acquisition and sustained fielding. The result is that potentially useful capabilities often do not mature into scalable, integrated systems that can be procured in quantity and employed broadly across the force.

Big “A” Capability Development and Current Reform

A second limitation lies in the DoD’s formal capability development enterprise. “Big A” acquisition traditionally referred to the interaction of the Joint Capabilities Integration and Development System (JCIDS), Planning, Programming, Budgeting, and Execution (PPBE), and the Defense Acquisition System (DAS), which together linked requirements definition, resourcing, and acquisition execution in order to move a capability from identified need to funded program and fielded system (Lofgren, 2019). Although intended to provide discipline and accountability, this construct often operated more in sequence than in parallel. The result was a process that could be orderly, but slow to adapt when capability definition, technical maturation, funding alignment, and transition planning needed to inform one another during development.

That historical construct is now being actively reworked. In November 2025, the Department directed major reform of the joint requirements process, including establishment of the Requirements and Resourcing Alignment Board (RRAB), the Mission Engineering and Integration Activity (MEIA), and the Joint Acceleration Reserve (JAR), while also directing removal of references to JCIDS from relevant policy and associated guidance. In parallel, the Department redesignated the DAS as the Warfighting Acquisition System and issued broader acquisition transformation guidance oriented toward speed, adaptation, and delivery of operational capability (Office of the Secretary of War, 2025a, 2025b; Office of the Under Secretary of War for Acquisition & Sustainment, 2025). At the strategic level, these actions acknowledge the same underlying problem addressed in this paper: capability development processes that are too sequential, too administratively rigid, and too disconnected from iterative technical learning. Even so, enterprise-level reform does not by itself specify how an individual development effort should be organized to operate differently in practice. That execution problem is where the embedded capability development model is intended to contribute.

The Embedded Capability Development Model

Embedded capability development is proposed here as a practical execution model for programs that must mature technical capability, refine requirements, and prepare for transition in parallel. As shown in Figure 1, the model occupies a middle ground between rapid, prototype-driven experimentation and formal capability development processes. Its distinguishing feature is the concentration of technical development, accountable requirements refinement, and transition alignment within a single continuously evolving effort rather than across sequential organizational handoffs.

Figure 1. Conceptual Depiction of the Embedded Capability Development Model

Early stakeholder engagement, iterative prototyping, digital engineering, and government-industry collaboration are all familiar to the defense development community (GAO, 2023, 2025; Modigliani et al., 2020). What distinguishes the present model is their integration within a single execution construct centered on an embedded government developer who is not merely a technical participant, but the principal government lead for day-to-day refinement of evolving operational and system-level requirements. In this construct, that individual combines responsibilities that are often distributed across separate actors: direct technical participation, accountable refinement of requirements, continuous alignment with transition-relevant stakeholders, and stewardship of a retained government baseline. This changes not only how information moves across the effort, but when, where, and by whom critical development decisions are made.

Organizational Structure

The organizational structure of the model is central to its function. For the model to work, the embedded developer must be operationally aligned within the chain of command responsible for capability development and requirements ownership while also operating inside a technical environment that supports direct design, build, test, and analysis. In this arrangement, formal authority over operational and system-level requirements remains with the appropriate capability development chain, but the embedded developer serves as the day-to-day integrating point through which technical execution, evolving requirements, and transition considerations remain connected. This does not consolidate every institutional function into one person or office. Rather, it preserves the formal requirements authority of the chain of command while establishing a clear point of continuity between technical development and the organizations responsible for requirement refinement, transition, and broader capability development.

Enabling Functionality

A second enabling feature of the model is the way it advances technical development, requirements refinement, and transition preparation together within the same effort. Rather than treating these as separate downstream activities, the model allows requirements to evolve alongside prototype evidence, keeps technical work tied to acquisition and integration realities, and generates the artifacts needed to support later transition and fielding. The result is not merely a working technical capability, but a more transition-ready one.

Core Drivers of Execution

In practice, the effectiveness of the model depends on several reinforcing features that shape execution.

1. **Continuous technical participation by the embedded developer.** The embedded developer does not operate solely as a sponsor, coordinator, or oversight actor. The role carries direct involvement in technical development, allowing requirement refinement to remain informed by real design, build, test, and analysis activity rather than by intermittent review alone.
2. **Shared technical artifacts that preserve learning and enable iteration.** The effort maintains common technical artifacts that capture architecture, design knowledge, interfaces, verification logic, and implementation lessons. These artifacts provide a durable basis for collaboration, reduce learning loss across organizational boundaries, and allow the evolving baseline to be refined with greater continuity.
3. **Iterative requirements refinement informed by evidence.** Requirements are not treated as fixed end-state constraints established before meaningful technical learning occurs. Instead, they are refined through design, build, test, and trade-space evidence so that operational need, technical feasibility, manufacturability, and transition relevance can inform one another before the baseline hardens.
4. **Early industry collaboration against an evolving baseline.** Industry is engaged early enough to shape the effort rather than merely receive it. This allows manufacturing realities, implementation constraints, subsystem tradeoffs, and alternative solution paths to inform the baseline while design freedom is still high.
5. **Persistent alignment with transition, test, and operational stakeholders.** Transition, test, and operational stakeholders are engaged while the effort is still learning. This ensures that downstream constraints and opportunities influence development early, reducing the likelihood that promising technical efforts later stall because key realities were surfaced too late.

Taken together, these drivers form the practical execution logic of the model. They explain why embedded capability development is more than a coordination construct. It is a way of organizing a program so that technical learning, evolving requirements, and transition preparation reinforce one another during execution rather than converging only after the fact.

Case Study: Low-Cost Tactical Missile Development

With the model established conceptually, this paper now turns to an ongoing case study in low-cost tactical missile development. Tactical missile development provides a useful test case because it combines high technical complexity, significant transition requirements, limited commercial sustainment incentives, and strong operational pressure for both speed and scale. These characteristics make it well suited for examining an approach intended to integrate requirement refinement, prototype maturation, and transition preparation within a single effort. If the model can add value in a domain as technically demanding and transition-heavy as tactical missile development, that strengthens the case for its applicability to other development-intensive military capabilities.

In the present case, the effort did not begin from a blank sheet. It built on a retained government technical foundation that had already established an initial system concept, generated flight-tested evidence, and captured early design and interface knowledge. That starting point shaped how the model was implemented in this program. It provided the initial basis from which later execution tools, including the government reference architecture and government reference design, could be developed and used to support collaborative refinement, technical maturation, and transition preparation. The discussion therefore begins with that baseline and the organizational context that enabled it before turning to the specific mechanisms through which the current effort is being executed.

Government-Owned Technical Baseline

A defining feature of the present case is that it began with a government-owned technical baseline established before the current program construct was formed. That baseline originated in a government-sponsored research effort conducted at the Naval Postgraduate School (NPS). Executed as the primary author’s doctoral dissertation, the effort examined lean, iterative approaches to the development of low-cost guided missile systems for future government capability development (Pierce, 2025). Although academically executed, the research was operationally motivated, sponsor-funded, and structured to generate reusable technical artifacts and insight for subsequent government efforts.

To establish that system-level missile baseline, the research employed an end-to-end design-build-test methodology that included the design, fabrication, and flight testing of a low-cost guided missile prototype. The purpose of the effort was not to produce a fielded weapon or acquisition-ready system, but to serve as an executable learning platform through which guidance, control, integration, and operational assumptions could be explored empirically. The resulting prototype, shown in Figure 2, provided a flight-testable starting point from which the later government-owned design baseline evolved. Data and observations from flight experiments informed the development of a government-owned reference architecture, initial interface assumptions, and an early understanding of system-level requirements. Retained as government-owned research outputs, these artifacts established a modular technical baseline structured to support experimentation, risk reduction, and requirements refinement independent of any specific vendor implementation or production pathway. As such, the effort provided both the technical foundation for the current case study and the basis for a follow-on organizational arrangement capable of carrying that baseline into continued collaborative development.

Figure 2. Exterior View of the Flight-Test Prototype Developed During the Prior Government-Sponsored Research Effort that Established the Initial Government-Owned Technical Baseline for the Current Effort

Implemented Organizational Structure

Building on that baseline, the implemented organizational structure for this case study is shown in Figure 3 and reflects a dual-hat arrangement intended to connect formal capability development authority with a hands-on technical development environment. This structure is enabled in part by the flexibility of the Marine Corps' PhD Program–Technical (PHDP-T), which allows technically trained officers to be assigned dynamically to high-impact roles aligned to Service needs (HQMC CD&I, 2022). In this case, the embedded capability developer remains operationally aligned to Headquarters Marine Corps Combat Development and Integration (CD&I), reporting through Marine Corps Science and Technology / Rapid Capabilities Office leadership and coordinating across the relevant capability development organizations, including operational analysis, requirements, and transition stakeholders. More broadly, this arrangement is consistent with the Marine Corps' movement toward tighter integration of requirements, acquisition, and capability development functions under the newly established PAE-MC construct.

At the same time, technical execution is conducted at NPS, where faculty appointment and research access provide the facilities, technical ecosystem, and collaborative environment necessary to support hands-on design, build, test, and analysis. Additional coordination across NPS organizations further supports research direction, innovation activity, and technical execution. Within this arrangement, the embedded capability developer serves as the connective element between operational authority, technical development, transition coordination, and external collaboration, enabling rapid development cycles while preserving alignment with broader Service capability priorities. This organizational arrangement, however, is only one enabling condition of the realized model. Its effectiveness also depends on sustained, early, and technically meaningful collaboration with industry during execution.

Figure 3. Organizational Realization of the Embedded Capability Development Model in the Current Case Study, Showing Command Alignment, Technical Execution Environment, and Collaborative Coordination Pathways

Collaborative Missile Development with Industry Partners

Complementing the organizational structure, a second key feature of the realized model is close technical collaboration with industry early in development. Once the initial government baseline had been established and feasibility demonstrated against an operationally relevant need, the effort was positioned for follow-on development within a broader naval capability program. It now operates as a deliberate, funded S&T effort focused on refining the technical baseline, interface definitions, subsystem performance parameters, and manufacturability considerations. In this phase, Office of Naval Research (ONR) sponsorship provides the resources necessary to move beyond the initial feasibility effort and support sustained development alongside structured industry participation.

Within that framework, collaboration with industry is conducted through Cooperative Research and Development Agreements (CRADAs) at NPS, allowing government and industry participants to work in close technical coordination while preserving government ownership of the broader technical baseline. As shown in Figure 4, this is not a one-way handoff from government requirements to industry execution. Rather, the government reference architecture (GRA), government reference design (GRD), and industry prototype path inform one another throughout execution. This allows prototype development, technical risk reduction, and requirements refinement to proceed together rather than through pre-stipulated handoffs. In this way, industry prototypes both inform and are informed by the evolving government baseline, allowing technical tradeoffs, interface definitions, and system parameters to be refined collaboratively before requirements are fully fixed. The practical mechanisms through which this collaboration is carried out are the GRA and GRD, which provide the shared technical basis for iterative requirements refinement, trade-space exploration, and prototype maturation.

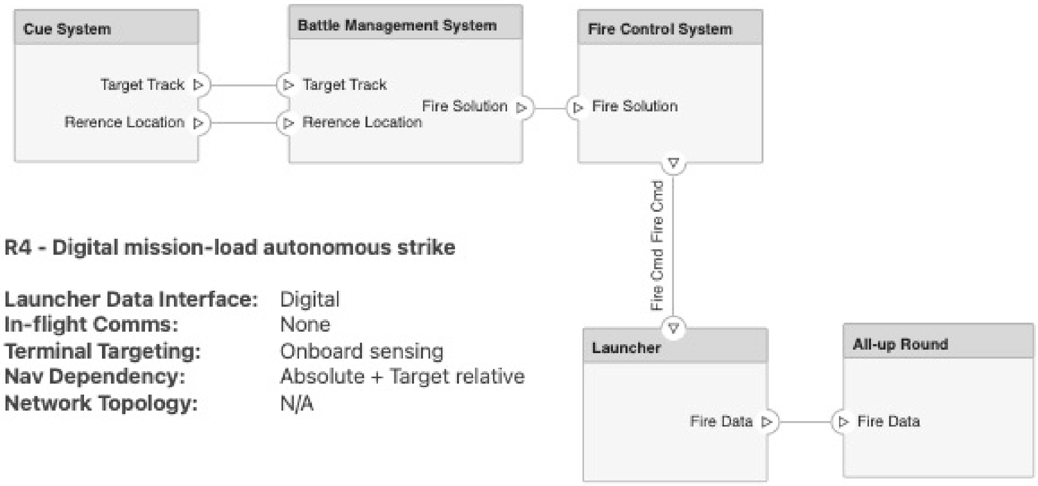
Figure 4. Dual-Path Development Structure Used in the Current Effort, Showing the Interaction Among the GRA, GRD, and Industry Prototype Path Within the ONR-Sponsored Development Effort

GRA

The first of these enabling tools is the GRA, defined as an authoritative government source of information that guides and constrains architectures and solutions within a specific subject area (Office of the DoD Chief Information Officer, 2010; Office of the Under Secretary of Defense for Research and Engineering, 2020). In traditional defense development, requirements and related baselines are often established early within broader sequential processes, while the technical learning needed to understand their downstream design, cost, schedule, and interface implications emerges later through development and testing. In the present approach, the GRA functions as a living development tool, evolving in parallel with the GRD and prototype evidence so that requirements, interfaces, and verification logic can be refined iteratively as technical understanding matures.

As a concrete example of the present implementation, Figure 5 depicts a high-level abstraction of one missile reference architecture developed in MATLAB System Composer. At this level, the architecture represents a mission thread for the missile system and is used to understand both the operational concept and the downstream system components required to realize it. Early in development, the GRA serves as a tool for trade-space analysis across alternative mission threads and operating concepts, allowing the team to evaluate how different approaches to achieving an operational effect drive different technical solutions and requirement implications.

The architecture shown in Figure 5 represents a digital mission-load autonomous strike concept in which the missile is preloaded from the launcher with targeting data, receives no inflight target updates, carries its own terminal guidance capability onboard, and is not networked to other systems. That architecture drives specific subsystem and interface implications, including the need for an onboard terminal seeker and a launcher-to-missile data interface. By contrast, an architecture based on continuous laser designation changes both the operational context and the downstream system requirements because it depends on an external designator throughout the missile's flight and therefore alters the required subsystem configuration. In this way, the GRA helps illuminate the broader trade space associated with closing the kill chain, including operational constraints, interface needs, and the likely cost and performance implications of different architectural choices. As the architecture and reference design mature together, the GRA evolves from a lower-fidelity trade-space tool into a higher-fidelity authoritative source for requirements, interfaces, and verification logic, while also supporting automated verification of digital government and industry prototype solutions. The result is greater understanding when design freedom is highest and more informed requirements and system decisions as the effort matures.



R4 - Digital mission-load autonomous strike

Launcher Data Interface: Digital
In-flight Comms: None
Terminal Targeting: Onboard sensing
Nav Dependency: Absolute + Target relative
Network Topology: N/A

Figure 5. High-Level Abstracted Example of a Missile GRA Illustrating One Mission-Thread Implementation

Although the embedded approach is not dependent on any single modeling tool, the choice of tooling materially affects how effectively the reference architecture and reference design can function as living development artifacts. In the present effort, the modeling environment is used not only to capture architectural relationships, but also to connect those relationships to higher-fidelity digital models and simulations associated with individual subsystems and prototype implementations. This allows the architecture to serve as more than a documentation product. It becomes part of an integrated digital ecosystem through which alternative configurations can be explored, prototype behavior can be assessed, and evolving requirements and interfaces can be verified against more realistic technical evidence.

GRD

While the GRA defines the structure of the solution space, the GRD provides an executable system baseline through which those architectural decisions can be implemented and tested. In the present effort, the GRD is a government-retained missile system design containing the artifacts needed to build, replicate, test, and refine the capability in practice. These artifacts include physical design files such as computer-aided design and printed circuit board layouts, flight software, and associated simulation capability. As a result, the GRD serves as a practical vehicle for technical risk reduction, low-cost knowledge capture, and preservation of design knowledge that might otherwise remain locked inside a vendor-owned implementation.

More importantly, the retained government baseline is not merely a technical convenience. It is a strategic enabler of continuity, competition, and transition. By preserving technical learning inside a government-owned reference design, the model reduces dependency on any single performer, improves the government’s ability to make informed transition decisions, and creates a stronger foundation for future competition, adaptation, and scaling. In this respect, the GRA and GRD support not only engineering development, but also continuity, transition planning, and future competition. Because the GRD is matured collaboratively with industry rather than in isolation, manufacturing realities, subsystem tradeoffs, and implementation constraints can shape design evolution early. In this way, the GRD helps generate more informed requirements, reduces the likelihood of late-stage transition failure, and provides a shared technical baseline through which government and industry can jointly burn down risk while preserving future flexibility.

As a concrete example of this coupling, Figure 6 depicts an abstracted portion of the missile flight software contained within the GRD. In this case, the software is developed in Simulink, auto-generated into flight code, and linked to a higher-fidelity six-degree-of-freedom simulation environment. This provides a direct connection between the evolving GRA and the underlying system realization. Interfaces represented in the flight software can be checked against the subsystem-to-subsystem contracts defined in the architecture, while alternative modules or subsystem implementations can be assessed against the same architectural requirements. In this way, the GRA and GRD are developed together rather than separately: the architecture informs the design, while the evolving design and simulation evidence refine the architecture. The result is a more integrated environment in which architecture, software, subsystem verification, and prototype maturation can converge together as the system develops.

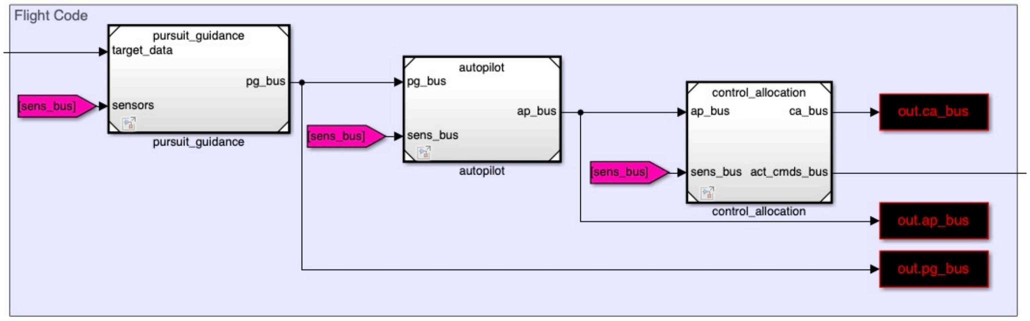


Figure 6. Abstracted Portion of the Model-Based Flight Software Within the GRD, Illustrating the Digital Implementation Layer Through Which Architectural Interfaces and Subsystem Contracts Can Be Refined and Verified

Execution of the Embedded Capability Development Model

The organizational structure, collaborative relationships, and enabling tools described above do not by themselves explain how the model functions in practice. In this program, the embedded capability development model is realized through the developer’s continuous involvement in technical design, requirements refinement, stakeholder coordination, and transition preparation. Rather than treating these activities as distinct phases or separate organizational handoffs, the embedded developer owns and integrates them within the same ongoing development effort. The following discussion highlights three practical dimensions of that execution: iterative requirements development, stakeholder integration, and technical coordination with industry.

Iterative Requirements Development

A central feature of the model in practice is that it begins from an operational need rather than from a fully specified set of operational and system-level requirements. In the present effort, the objective is not to translate a fixed requirement set downward into design, but to allow requirements to emerge and mature alongside design, build, test, and analysis activity. This reflects a different starting logic. Instead of assuming that the key system requirements are already known in sufficient detail, the effort treats many of them as matters to be discovered through technical development, trade studies, and empirical evidence. In this way, the model does not reduce requirements discipline; it relocates it into the learning process, where performance, manufacturability, cost, schedule, and operational utility can inform one another before the baseline hardens.

Figures 7 and 8 illustrate the distinction between the legacy linear requirements logic and the approach used in the present effort. Figure 7 depicts the historical deliberate requirements process as a reference point for the type of sequencing this paper seeks to overcome, while Figure 8 shows the coupled refinement process used here. In the traditional model, an operational need is translated relatively early into formal requirements, often before the technical implications of those requirements are fully understood. In the present approach, the operational need remains the starting point, but the detailed requirements needed to satisfy it are allowed to develop in parallel with the evolving technical baseline. Design evidence, build-test feedback, and implementation realities therefore shape the resulting requirement set rather than merely test compliance against an early assumption. The result is a process that is less dependent on prematurely fixed system definitions and more grounded in demonstrated knowledge about what the system can do, what tradeoffs are required, and what level of performance is realistically achievable to satisfy the operational need.

A concrete manifestation of this approach in the present effort is the use of set-based analysis before formal requirement lock. The program begins with a high-level operational need rather than a fully stipulated system definition, and alternative solution paths are explored against factors such as technical feasibility, manufacturability, integration burden, market availability, and transition relevance. In parallel, focused prototype efforts with the GRD are used to retire selected technical risks and capture key knowledge points. Those results feed back into the evolving baseline, allowing requirements to emerge from demonstrated evidence and informed trade decisions rather than from early assumptions about what the final system should be.

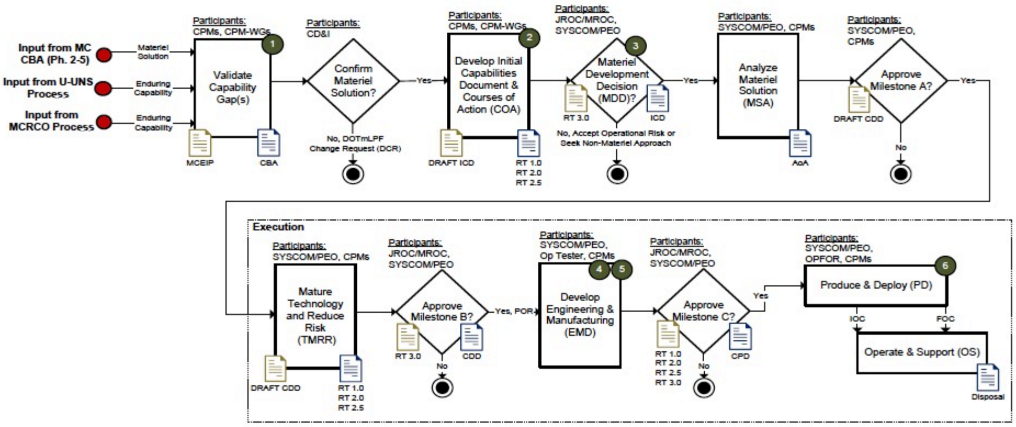


Figure 7. JCIDS Deliberate Requirements Process. Adapted from Force Development System User Guide (United States Marine Corps, 2018).

Figure 8. Iterative Requirements Refinement Process Used in the Present Effort, Showing How Operational Need, Build-Test Feedback, Manufacturing Feedback, and Design Feedback Inform Coupled Operational and System Requirements Captured in the Digital Environment

Early Stakeholder Integration and Transition Alignment

A second practical dimension of the model is the deliberate early engagement of stakeholders that are often brought in only after a prototype has already matured. In the present effort, the embedded capability developer serves as the integrating point through which those organizations are pulled into the development process while the baseline is still evolving. As depicted conceptually in Figure 9, this includes the S&T sponsor, represented by ONR; Service capability development stakeholders within Headquarters Marine Corps CD&I; potential transition stakeholders within Program Executive Office Land Systems; operational test stakeholders within the Marine Corps Operational Test and Evaluation Activity; the technical execution environment at NPS; supporting technical capacity from University Affiliated Research Centers, Federally Funded Research and Development Centers, and Warfare Centers; and collaborating industry participants engaged through CRADAs.

Figure 9. Conceptual Depiction of the Embedded Capability Developer as the Integrating Node for the Stakeholder Network Required to Mature the Capability Beyond a Standalone Prototype

The purpose of this early integration is not simply to improve communication. It is to surface downstream realities early enough to shape the evolving baseline. Program office and systems engineering stakeholders can help ensure alignment with existing program trajectories, launcher and interface realities, and broader transition constraints. Operational test stakeholders can help identify opportunities to retire relevant test objectives during development rather than after the fact. Capability development and resourcing stakeholders can help keep the effort aligned with operational priorities and future programmatic considerations. In this way, the model pulls traditionally downstream considerations into the active development effort, reducing the likelihood that a technically promising prototype later stalls because key constraints, interfaces, or transition expectations were discovered too late.

Technical Collaboration with Industry Partners

A third practical dimension of the embedded capability development model is the manner in which technical coordination with industry is carried out during execution. In this program, the embedded capability developer does not engage industry primarily through episodic reviews or formal handoff points, but through sustained day-to-day interaction within a shared technical development rhythm. This includes common access to relevant development artifacts and environments, such as software repositories, model-based design products, computer-aided design files, and interface definitions, together with recurring sprint-style engagements used to identify near-term objectives, technical blockers, and risk-burndown priorities.

What distinguishes this arrangement is that the embedded capability developer participates directly in low-level technical development while simultaneously interpreting those decisions against evolving operational requirements, transition constraints, and broader stakeholder priorities. This interaction is grounded not merely in recurring coordination, but in shared access to the evolving technical baseline, allowing discussion, trade studies, and implementation changes to occur against the same architecture, design, and interface context. Equally important, the collaboration framework reduces many of the practical administrative barriers that often slow government-industry interaction, including access to development environments, site access, and security-related coordination. The result is a form of collaboration that compresses technical, requirements, and transition feedback loops into the same development rhythm, reducing translation loss across organizational boundaries and allowing government and industry to work side by side on technical maturation while preserving alignment with the larger capability development effort.

Early Observations from Implementation

Although the effort remains ongoing, several practical observations can already be drawn from its implementation. These do not yet constitute a full assessment of transition outcomes, but they do show how the model is changing the development process in practice. Two examples are particularly illustrative: tighter coupling of architecture, design, and verification within the government baseline, and the use of set-based analysis before formal requirement lock.

One concrete example of execution impact in the present effort has been the maturation of the GRD’s software and simulation environment through close collaboration with an industry partner. Earlier versions of the government baseline relied on a more monolithic flight software structure that was only loosely connected to the higher-fidelity simulation environment. Through collaborative development, that baseline was reworked so that model-based flight software modules could be auto-generated from simulation and exercised within the evolving implementation environment. This changed more than software development efficiency. It created a tighter connection between the GRD and the GRA, allowing digital interfaces, latency assumptions, navigation error parameters, and other subsystem-level considerations to be evaluated against a more realistic system realization earlier in development. In practical terms, this improved the quality of the government baseline itself by making later modular verification more credible and by allowing both government and industry implementations to be assessed against the same evolving architectural contracts and program objectives.

A second example is the use of set-based analysis before formal requirement lock. Rather than beginning with a fully stipulated system definition, the program starts from a high-level operational need and explores multiple feasible solution paths against factors such as technical feasibility, manufacturability, integration burden, market availability, and transition relevance. In parallel, focused prototype work with the GRD is used to retire selected technical risks and capture key knowledge points. The practical effect has been to shift requirement development away from early assumption and toward evidence-based convergence. Instead of treating requirements as fixed inputs to design, the present effort has allowed them to emerge from demonstrated trade studies, prototype results, and implementation realities. This has improved the technical grounding of the evolving requirement set while preserving flexibility during the early stages of development.

Taken together, these examples illustrate the practical value of the embedded approach in its current stage of execution. The benefit is not yet a completed transition outcome, but a stronger development baseline: one in which architecture, design, requirements, and transition considerations remain coupled while the system is still being defined. That integrated baseline should improve the quality of later transition decisions and reduce the likelihood of avoidable rework as the effort matures.

Conditions for Applicability

The embedded capability development model is not a universal replacement for traditional acquisition, nor is it the only viable approach for accelerating capability delivery. It is best understood as one execution model within a broader set of emerging efforts to increase speed, flexibility, and operational relevance in defense development. Its greatest utility lies in a specific class of problem: capabilities that require meaningful technical maturation, involve nontrivial integration and transition complexity, and cannot be responsibly defined or acquired through either pure commercial purchase or a simple compete-then-commit logic at the outset.

First, the model is best suited to development-intensive capabilities that require genuine technical maturation and military integration but remain bounded enough to be advanced by a relatively agile technical effort. It is less relevant for pure commercial-off-the-shelf procurement, where the government can function primarily as a buyer, and less suited to very large, capital-intensive programs that require long-horizon program structures and extensive formal oversight from inception. Its natural fit lies between those extremes: efforts that must still be engineered, tested, and integrated, but that benefit from rapid iteration and tighter coupling between technical learning and capability definition.

Second, the model is most useful when the operational need is clear but the technical solution remains meaningfully uncertain. In such cases, architecture, subsystem selection, implementation constraints, manufacturability, and transition considerations must be learned together rather than assumed in advance. This distinguishes the model from approaches in which industry can be asked to build first against a relatively stable need and then compete for later production. Where the central challenge is not simply buying or selecting among candidate solutions, but discovering what the right requirement set should be as the system matures, embedded capability development offers a more appropriate mechanism.

Third, the model depends on an appropriately positioned government lead. The embedded developer must combine technical fluency, operational understanding, and sufficient institutional access to connect technical execution with broader capability development priorities. The role is especially well suited to personnel who can operate credibly across both engineering and operational communities while remaining nested within the chain of command responsible for capability development and formal requirements ownership. Without that alignment, the model risks losing the continuity that gives it value.

Finally, the model requires both a viable development environment and a plausible transition pathway. It depends on access to an environment capable of supporting real design, build, test, and analysis activity, together with the institutional relationships needed to carry a successful effort forward if the technical work proves promising. Without those conditions, the model risks becoming either a coordination construct without technical leverage or a prototype effort without a realistic path beyond demonstration.

Risks and Limits

The model also carries important limits and risks. First, it depends heavily on personnel quality, institutional trust, and the availability of collaborative development authorities that may not exist in every program. If the embedded government lead lacks sufficient technical credibility, operational understanding, or institutional access, the model can quickly collapse into coordination without real developmental influence. Second, the model can create ambiguity if decision authority, government ownership boundaries, and transition expectations are not established clearly at the outset, particularly in a collaborative environment involving industry, sponsors, and potential transition stakeholders. There is also a risk of overextension if the embedded capability developer is treated as a substitute for the full requirements, acquisition, or test communities rather than as an integrating point across them. Third, while the model can reduce technical and transition risk, it does not eliminate the need for formal acquisition discipline, nor does it by itself solve the industrial-base, production-capacity, or long-term sustainment challenges associated with fielding capability at quantity.

A further limitation is that the strongest evidence for the model's value will often emerge only in retrospect. Because the present case remains ongoing, the observations offered here are necessarily preliminary. The paper therefore does not claim that the model has already proven universal success. It argues instead that the current effort demonstrates a plausible and potentially valuable way to integrate development, requirements refinement, and transition preparation more closely than is typical in traditional approaches.

Implications for Capability Development

Taken together, the observations, conditions, and risks discussed above suggest that embedded capability development is best understood as a practical way to organize collaborative government-industry development for capabilities that cannot simply be bought and cannot be responsibly specified in full detail at the outset. Its value lies in coupling technical learning, evolving requirements, industry realities, and transition considerations while the capability is still being defined. In that sense, it shifts the effort away from requirements development for its own sake and toward a more grounded process in which requirements emerge from design evidence, trade studies, implementation constraints, and operational need.

The result should be a better-informed capability baseline: one shaped not only by what is operationally desirable, but also by what is technically feasible, manufacturable, testable, and realistically transitionable. That does not replace formal acquisition. It helps set acquisition up better by improving the quality of the technical and programmatic baseline before later commitments harden.

Recommendations

Based on the early execution of the model in the present case study, several preliminary recommendations emerge for applying embedded capability development more broadly.

1. **Apply the model selectively to development-intensive capabilities that cannot simply be bought and cannot be responsibly specified in full detail at the outset.** The model is most valuable where technical realization, operational utility, and transition considerations must be learned together rather than assumed in advance.
2. **Place the embedded developer inside both the technical development effort and the capability development chain responsible for requirements refinement.** The model depends on a lead who can participate directly in design, build, test, and analysis while maintaining continuity with the organizations responsible for evolving requirements, transition, and broader capability priorities.
3. **Use government-owned reference architectures and reference designs to support early collaborative development with industry.** Shared technical baselines provide a practical means of reducing risk, preserving government knowledge, improving requirements quality, and allowing government and industry to work against a common evolving baseline rather than through loosely coupled handoffs.
4. **Pull transition-relevant stakeholders and constraints left into development.** Early engagement with programmatic, systems engineering, test, and capability development stakeholders helps surface interface realities, transition constraints, and opportunities for risk retirement while the baseline is still evolving.

Conclusion

This paper presented embedded capability development and examined its implementation in an ongoing low-cost tactical missile development effort. The model is intended for capability problems in which technical development, evolving requirements, and transition preparation must proceed together rather than sequentially. In the present case, that approach has been realized through a retained government baseline, a dual-hat organizational structure, close government-industry technical collaboration, and the use of evolving reference artifacts to support iterative development.

It is still too early to make definitive claims about ultimate transition success or the model's broader applicability. Even so, early evidence from the present effort suggests that embedded capability development provides a feasible way to organize collaborative government-industry development for capabilities that cannot simply be bought and cannot be responsibly defined in full detail at the outset. Its value lies in allowing technical learning, evolving requirements, industry realities, and transition considerations to shape one another while the capability is still being defined. For efforts of this kind, that may be its principal contribution: not as a substitute for acquisition, but as a practical way to produce a more coherent, better-informed, and more transition-ready baseline before later commitments harden.

References

DoD. (2007, August). *Tech transition report to Congress* [Report to Congress]. https://www.dau.edu/sites/default/files/Migrated/CopDocuments/Tech%20Transition%20Report%20to%20Congress%20August%202007.pdf

DoD. (2020, September 9). *Department of Defense Directive 5000.01, The Defense Acquisition System* (Incorporating Change 1, July 28, 2022). https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodd/500001p.pdf

GAO. (2015, November). *Defense advanced research projects agency: Key factors drive transition of technologies, but better training and data dissemination can increase success* (GAO-16-5). https://www.gao.gov/products/gao-16-5

GAO. (2016, November). *Weapon system requirements: Detailed systems engineering prior to product development positions programs for success* (GAO-17-77). https://www.gao.gov/assets/gao-17-77.pdf

GAO. (2023, February 7). *Middle-tier defense acquisitions: Rapid prototyping and fielding requires changes to oversight and development approaches* (GAO-23-105008). https://www.gao.gov/products/gao-23-105008

GAO. (2024, December 12). *DOD acquisition reform: Military departments should take steps to facilitate speed and innovation* (GAO-25-107003). https://www.gao.gov/products/gao-25-107003

GAO. (2025, June 11). *Defense acquisition reform: Persistent challenges require new iterative approaches to delivering capability with speed* (GAO-25-108528). https://www.gao.gov/products/gao-25-108528

Headquarters, United States Marine Corps, Combat Development and Integration. (2022, August 19). *Dynamic employment model decision paper: Marine Corps Doctor of Philosophy technical program dynamic employment model* [Decision paper].

Lofgren, E. (2019, April 16). *What do we mean by big "A" and little "a" in acquisition?* Acquisition Talk. https://acquisitiontalk.com/2019/04/what-do-we-mean-by-big-a-and-little-a-in-acquisition/

Modigliani, P., Ward, D., Lewis, T., & McGee, W. (2020, March). *Modernizing DoD requirements: Enabling speed, agility, and innovation*. MITRE Center for Technology and National Security. https://www.mitre.org/sites/default/files/2021-11/prs-19-03715-2-modernizing-dod-requirements-enabling-speed-agility-and-innovation.pdf

Murray, K. (2023, March). The Sea Services must build a new acquisition assembly line. *Proceedings, 149*(3/1,441). https://www.usni.org/magazines/proceedings/2023/march/sea-services-must-build-new-acquisition-assembly-line

Office of the Department of Defense Chief Information Officer. (2010, June). *DoD reference architecture description*. Department of Defense. https://dodcio.defense.gov/Portals/0/Documents/Ref_Archi_Description_Final_v1_18Jun10.pdf

Office of the Secretary of War. (2025a, November 7). *Reforming the joint requirements process to accelerate fielding of warfighting capabilities* [Memorandum]. https://media.defense.gov/2025/Nov/10/2003819442/-1/-1/1/REFORMING-THE-JOINT-REQUIREMENTS-PROCESS-TO-ACCELERATE-FIELDING-OF-WARFIGHTING-CAPABILITIES.PDF

Office of the Secretary of War. (2025b, November 7). *Transforming the Defense Acquisition System into the Warfighting Acquisition System to accelerate fielding of urgently needed capabilities to our warriors* [Memorandum]. https://media.defense.gov/2025/Nov/10/2003819439/-1/-1/1/TRANSFORMING-THE-DEFENSE-ACQUISITION-SYSTEM-INTO-THE-WARFIGHTING-ACQUISITION-SYSTEM-TO-ACCELERATE-FIELDING-OF-URGENTLY-NEEDED-CAPABILITIES-TO-OUR-WARRIORS.PDF

Office of the Under Secretary of Defense for Research and Engineering. (2020, November). *Mission engineering guide*. Department of Defense. https://ac.cto.mil/wp-content/uploads/2020/12/MEG-v40_20201130_shm.pdf

Office of the Under Secretary of War for Acquisition & Sustainment. (2025, November 10). *Acquisition transformation strategy: Rebuilding the arsenal of freedom* [Report]. https://media.defense.gov/2025/Nov/10/2003819441/-1/-1/1/ACQUISITION-TRANSFORMATION-STRATEGY.PDF

Pierce, D. T. (2025). *Capability-cost inversion: An iterative framework for the development and flight-test validation of low-cost guided missiles* [Doctoral dissertation, Naval Postgraduate School, limited distribution].

United States Marine Corps. (2018, April). *Force development system user guide*. Headquarters, United States Marine Corps.

Naval
Postgraduate
School
Foundation

Acquisition Research Program
Department of Acquisition, Finance, and Manpower
Naval Postgraduate School
555 Dyer Road, Ingersoll Hall
Monterey, CA 93943

www.acquisitionresearch.net