



RFI RESPONSE - CONFIDENTIAL

VIRTUAL MAINTENANCE TRAINERS

Response to BAE Systems RFI BAE-LT-ACE-5516



PREPARED FOR

BAE Systems
Air Sector

POINT OF CONTACT

Stephanie Fox
Sr. Procurement
Engineer

RFI REFERENCE

BAE-LT-ACE-5516
Virtual Maintenance
Trainers

RESPONSE DATE

5 May 2026



COMPANY PROFILE

Legal Name	ForgeFX, Inc.
Headquarters	Sacramento, California, USA
Year Founded	2002
Team Size	28 members
RFP Contact	Mary Pierce, Executive Director of Enterprise Partnerships mary.pierce@forgefx.com 415-788-5725, Ext. 703

ForgeFX Simulations is a U.S.-based immersive training and simulation company with over 24 years of experience developing custom, virtual training systems for complex, safety-critical operations across government and commercial sectors.

ForgeFX solutions are built on the proprietary **ForgeSIM™** framework, a modular simulation platform that supports high-fidelity 3D environments, structured scenario variability, performance tracking, and instructor-led training workflows. The platform enables the development of scalable training systems that can expand over time without re-architecting core functionality.

ForgeFX delivers **platform-agnostic solutions** with no per-seat licensing or usage fees. Applications can be deployed across standalone VR, desktop simulators, and mixed physical-virtual environments, supporting flexible rollout across enterprise and training organizations. ForgeFX systems are currently deployed in production environments supporting enterprise and government training programs.

KEY CAPABILITIES

- Custom VR training simulator design and development (utilizing Unity or Unreal)
- No per-seat licensing; full client IP ownership
- Modular scenario logic with configurable training variability
- Equipment-accurate interaction models and physics
- Instructor-led and multi-user training workflows
- Real-world control (RWC) integration
- Integration with LMS and enterprise training systems
- Industry-agnostic architecture supporting industrial, energy, aviation, manufacturing, and defense

SUPPORTED DEVICE ECOSYSTEMS

STAND-ALONE VR
Meta Quest, Pico, HTC Vive XR

PC-TETHERED VR
HTC Vive, Varjo, Valve Index

DESKTOP SIMULATORS
Keyboard, mouse,
joystick-based systems

MIXED REALITY & PHYSICAL INTEGRATION
Custom hardware and
real-world control systems

COMPLIANCE SNAPSHOT

This table summarizes compliance at a high level, with detailed responses provided in the sections that follow.

Ref.	Requirement Area	Status	Evaluator Note
2.01	Modular Architecture	YES	Detailed evidence described on page 4
2.02	Behavioral Modelling & Data	YES	Detailed evidence described on page 5
2.03	User Interface & Interaction	YES	Detailed evidence described on page 6
2.04	Deployment & Infrastructure	YES	Detailed evidence described on page 7
2.05	Security & Compliance	PARTIAL	Cleared-prime partner model for work above current direct clearance tier. Detailed evidence described on page 8.
2.06	Build Reproducibility	YES	Automated, version-controlled builds with manifests and rollback support. Detailed evidence described on page 9.
2.07	Maintainability	YES	Detailed evidence described on page 10.
2.08	Performance & Scalability	YES	Detailed evidence described on page 11.
2.09	User & Operational Requirements	YES	Detailed evidence described on page 12.
2.10	Learner Experience	YES	Detailed evidence described on page 13.



The eight requirement areas below correspond to the core platform requirements named in the RFI. Each block restates the requirement in full, then carries the four-part response: claim, proof, technical details, and assumptions.

2.01 MODULAR ARCHITECTURE

The solution shall:

- Provide a core runtime platform configurable into multiple trainers
- Support clear separation between:
 - Core platform functionality
 - Trainer-specific behavior, content, and configuration
- Enable reuse without rework across multiple training variants

CLAIM

Yes. ForgeFX has built and delivers the ForgeSIM core framework, a configurable runtime that supports multiple training modules from a single, version-managed codebase. New trainers are delivered as modules that integrate with this shared runtime, allowing expansion of a training suite over time.

The core framework is designed to remain stable as new modules are added, with most development effort focused on trainer-specific behavior, content, and configuration. While minor extensions to the core may be required as new capabilities are introduced, the overall architecture minimizes rework and supports efficient reuse across training variants.

PROOF

This architecture is in active use with customers such as John Deere, where each new machine is delivered as an additional training module integrated into an existing runtime. In the Deere Multi-Machine Walkaround Inspection Trainer, a single ForgeSIM runtime supports multiple heavy-equipment modules, including a motor grader, excavator, dozer, and dump truck, all sharing a common inspection structure. As the training suite has expanded, new modules have been added using the same core runtime, with updates primarily scoped to machine-specific content and behavior rather than changes to the underlying platform.

TECHNICAL DETAILS

The ForgeSIM core framework is built on industry game engines and provides shared systems for rendering, input handling, interaction logic, telemetry, and user interface management. These systems are implemented in the core platform, while trainer-specific behavior, content, and configuration are organized as modular components layered on top. This separation allows a single codebase to support multiple trainers, with new modules introduced through content and configuration updates rather than changes to core systems. The approach supports consistent user experience, maintainability, and incremental expansion of training programs over time.

ASSUMPTIONS

The claim of reuse without rework assumes that new trainers use modalities already supported by the ForgeSIM platform (e.g., desktop, touchscreen, XR). Requirements that introduce new modalities or hardware integrations would require extension of the core platform, with that effort scoped separately from trainer content development.

2.02 BEHAVIORAL MODELING & DATA

The solution shall:

- Support authoring, execution, and governance of system behavior models
- Maintain a structured state model representing simulation variables
- Enable traceability between requirements, behavior, and outputs

CLAIM

Yes. ForgeFX represents system behavior using modular, data-driven components that can be configured and extended over time. Subject-matter experts contribute to behavior definition through structured inputs and review workflows, supported by versioning and controlled updates. Simulation state is organized in a structured format, with defined variables and inputs driving system behavior. Behavior outputs can be linked back to source data and design intent, supporting traceability between requirements, implemented behavior, and observed results. This traceability is maintained through version-controlled assets and documented configuration.

PROOF

A representative example is ForgeFX's work on the **MRIGlobal CBRND HoloTrainer**, where virtual chemical-detection sensors respond to data-driven simulated inputs. For example, a simulated plume of agent influences detector outputs based on defined response parameters, allowing the system to reflect expected device behavior. Subject-matter experts define expected input-output relationships through structured data inputs, and the simulator uses these definitions to drive runtime behavior.

TECHNICAL DETAILS

Behavior is implemented as modular, data-driven components, with configuration managed through structured editing tools and version-controlled assets. Changes are reviewed and validated through defined workflows prior to deployment. The simulation state is structured and supports a range of variable types, including discrete, continuous, and environment-driven inputs. Runtime sessions can be recorded and replayed to support review, testing, and iterative refinement. Where required, behavior definitions and configuration data can be associated with source requirements or design documentation, supporting traceability for testing and validation activities, including use in V&V processes.

ASSUMPTIONS

The effectiveness of SME-driven authoring depends on the availability and completeness of source data, such as schematics, control logic, and fault conditions. ForgeFX's process supports phased development to accommodate partial or evolving inputs; however, the fidelity of the final model is directly dependent on the quality of the provided source material.

2.03 USER INTERFACE & INTERACTION

The solution shall:

- Provide a modern, scalable UI framework
- Support:
 - Desktop/classroom use
 - Touch-enabled interfaces
 - XR (VR/AR/MR) where appropriate

CLAIM

Yes. ForgeFX employs a component-based UI framework that supports delivery across desktop, touch, and XR modalities. Core training content and interaction logic are authored once and deployed across multiple platforms. UI presentation and input methods are adapted to the target device (e.g., mouse and keyboard for desktop, touch for tablets, and controller or hand interactions for XR), allowing the same underlying training experience to be delivered without re-authoring core content.

PROOF

ForgeFX's simulation work on **JLG Boom Lift, Scissor Lift, and Telehandler** trainers demonstrates this approach. The same training application has been deployed across multiple modalities, including desktop, iPad (touch), and VR with real-world hardware controls.

Core content and training logic are shared across deployments, while UI presentation and interaction methods are adapted to each platform.

TECHNICAL DETAILS

The platform supports multiple interaction modalities, including desktop, touch-based interfaces, kiosk-style deployments, and XR headsets. Training procedures are authored once and structured to allow reuse across these delivery formats, with presentation and interaction layers adapting to the target environment. Interaction handling is abstracted to support different input types, enabling consistent execution of training logic across mouse/keyboard, touch, and XR controller or hand-based inputs. Shared assets and content formats are used to maintain consistency and reduce duplication across platforms. Accessibility considerations, such as WCAG 2.1 AA alignment where applicable, are incorporated into UI design and implementation.

ASSUMPTIONS

Current XR support includes commonly used platforms such as Meta Quest, Pico, HTC Vive, and HoloLens. Integration of a new or unsupported device would require additional device-specific development effort, scoped separately. Core training content and logic would remain unchanged, with integration focused on adapting input handling and device-specific interaction.



2.04 DEPLOYMENT & INFRASTRUCTURE

The solution should support:

- Flexible deployment models, including:
 - Standalone (air-gapped)
 - On-premises networked
 - Centrally managed / hosted (where permitted)

CLAIM

Yes. ForgeFX supports standalone (air-gapped), on-premises, and centrally hosted deployment models. Builds for each model are produced from a common codebase, allowing the same core application to be deployed across different environments. While packaging and configuration may vary by deployment target, the underlying content and training logic remain consistent. This allows customers to adjust deployment strategies over time without requiring a full re-authoring effort.

PROOF

ForgeFX has delivered training applications across all three deployment models. For example, a standalone version of **Halliburton's Hydraulic Workover VR** trainer is deployed in offshore environments with no outbound network access. On-premises deployments, such as **StrataTech's OcuWeld** welding trainer, are hosted on school-managed networks with authentication integrated through Canvas LMS. Centrally hosted applications, including the US Army's Jacks **Unmanned Aerial System** trainer, are delivered as browser-based experiences hosted on a managed platform.

TECHNICAL DETAILS

All three deployment models are supported from a common codebase, with build configurations determining the final delivery format. Browser-based deployments are delivered as WebGL applications accessible via URL with no local installation required. Standalone deployments are provided either as packaged desktop applications or hosted on a local machine within an isolated network. On-premises deployments are typically hosted on customer-managed infrastructure, such as an LMS or internal web server, with authentication and access control handled by the customer environment. Centrally hosted deployments are delivered through a managed cloud environment operated by ForgeFX or a hosting partner.

ASSUMPTIONS

Centrally hosted deployments are provided by default from a US-region Microsoft Azure environment. Programs requiring a sovereign or government-compliant environment can be deployed to a cleared hosting provider as needed. For standalone (air-gapped) deployments, features that depend on external network services are disabled or replaced with local equivalents to ensure full offline operation.

2.05 SECURITY & COMPLIANCE

The solution shall be capable of operating under defense security and classification constraints

CLAIM

ForgeFX delivers training applications that operate within controlled environments, including Controlled Unclassified Information (CUI), For Official Use Only (FOUO), and similar categories. ForgeFX has not performed development within Secret-classified environments. For programs requiring higher classification levels, ForgeFX works in coordination with a cleared prime contractor or hosting partner that provides the required personnel clearances and secure facilities. In these cases, ForgeFX's scope is limited to simulation development, while the prime contractor maintains responsibility for the overall security environment and compliance posture.

PROOF

ForgeFX has supported defense-related programs in partnership with cleared prime contractors. For example, ForgeFX contributed to MRIGlobal's work under JPEO-CBRND, where ForgeFX developed simulation components for the **CBRND HoloTrainer** and related systems, while MRIGlobal operated as the cleared prime contractor. ForgeFX has also supported programs with Leidos on a DARPA-funded effort and with Applied Research Associates (ARA) on a DTRA program, providing simulation development within government-directed program environments. In addition, ForgeFX supported the US Army **UAS Tele-Op Simulator** port to the Jacks platform, providing simulation development within government-directed program environments, including work aligned with controlled and restricted data environments.

TECHNICAL DETAILS

ForgeFX applications are designed to support deployment into controlled, restricted, and air-gapped environments. Delivered training applications are generated without embedded credentials. Telemetry and external data transmission are disabled by default unless explicitly enabled. Sensitive configuration, such as API keys or access credentials, is externalized and injected at build or runtime to align with customer-specific security requirements.

Applications are designed to operate in offline or restricted network environments, with no dependency on external services unless explicitly configured. This allows deployment into controlled or air-gapped environments without requiring changes to core functionality. Integration with identity, access control, and network security is handled in coordination with the customer or prime contractor environment. ForgeFX applications are built to integrate with customer-managed identity providers and security controls where required, without introducing additional external dependencies.

ForgeFX follows established configuration management and secure development practices aligned with SOC 2 Type 1 readiness. For programs requiring higher classification levels, secure hosting, accreditation, and network controls are provided by the prime contractor or designated hosting partner.

ASSUMPTIONS

ForgeFX is a US-based company, and its development team does not currently maintain personnel with active Secret or Top Secret security clearances. Requirements involving cleared personnel, secure facilities, or classified environments are addressed through partnership with a cleared prime contractor or designated direct hosting provider. Programs requiring direct ForgeFX personnel access to classified environments would require those personnel to obtain the appropriate security clearances.

2.06 BUILD REPRODUCIBILITY

The solution shall enable repeatable, version-controlled builds

CLAIM

Yes. Trainer builds are produced through a standardized, automated pipeline using a single Git-based source of truth. Each build is tied to a specific commit and includes a recorded set of key version information such as engine version, project dependencies, and included content.

Build outputs are archived with their associated metadata, allowing prior versions to be re-created under controlled conditions. This supports auditability, rollback to known-good versions, and maintenance of multiple trainer variants over time.

PROOF

ForgeFX production workflows use a consistent CI/CD approach across projects, with Git for source control and automated build processes for packaging and deployment. Each release is traceable to a specific source commit and can be redeployed as needed.

This approach has been applied across multiple customer programs, including John Deere, Halliburton, StrataTech, and the US Army, supporting consistent delivery and update management across different training applications.

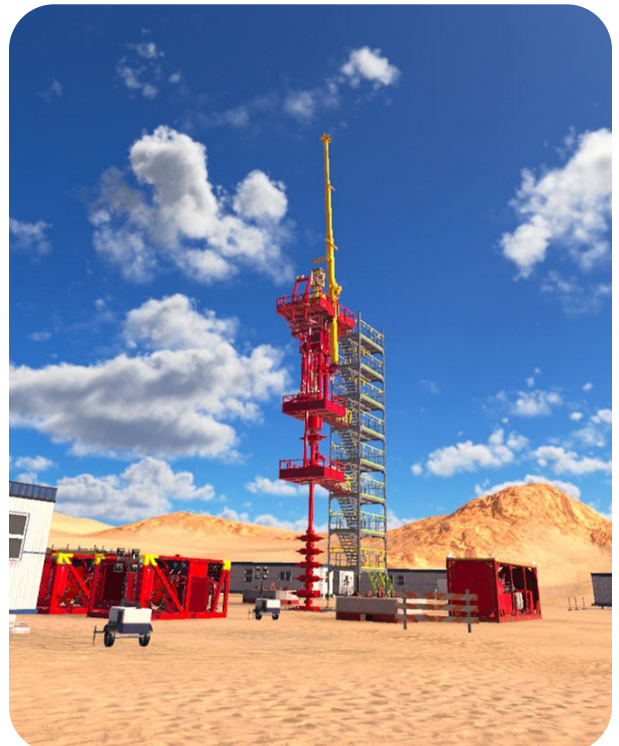
TECHNICAL DETAILS

Git serves as the system of record for all source content, with all code, assets, and configuration managed through version control. Automated build pipelines generate application builds from defined branches or tagged versions, ensuring consistency across releases. Each build artifact is stored along with its associated metadata, including engine version, dependency versions, and build configuration.

Dependencies are managed and updated in a controlled manner to reduce drift between builds and maintain compatibility over time. Prior builds can be re-generated or redeployed using the archived source and configuration data, supporting rollback to known-good versions when needed. The overall release process is structured to support the maintenance and update of multiple deployed trainer versions in parallel.

ASSUMPTIONS

None



2.07 MAINTAINABILITY

The solution should:

- Reduce reliance on specialist knowledge
- Enable a scalable developer and support workforce

CLAIM

Yes. ForgeFX's authoring tools are designed to support subject-matter experts (SMEs) and instructional designers in managing training content, while maintaining a clear separation between content updates and core system functionality. Routine updates, such as adjusting lesson content or parameters, can be performed through guided tools, while more advanced changes are supported by a modular codebase, automated testing, and documented workflows. This structure reduces reliance on specialized knowledge and supports long-term maintainability. ForgeFX provides a structured handoff of source code, build environments, and authoring tools, enabling customers to transition to an in-house sustainment model if desired.

PROOF

ForgeFX's work with **Vermeer** demonstrates this model. Following initial development and delivery, ForgeFX transferred the full source code and supporting tools to Vermeer's internal team. With documentation and onboarding support, Vermeer's developers were able to take over sustainment activities and implement updates independently. This reflects a successful transition to a customer-led, in-house maintenance model.

TECHNICAL DETAILS

Training content is authored through guided tools aligned with SME and instructional design workflows. Content configuration is separated from application logic through structured data and editor-driven workflows. This separation allows content updates to be performed without modifying underlying systems. The codebase is delivered with documentation and structured to support maintainability, including modular components and defined workflows for updates and testing. Assets are managed using standard formats to support reuse and compatibility across tools. A standard handoff includes source code, build environments, and authoring tools, supported by a structured onboarding period to prepare the receiving team for ongoing maintenance.

ASSUMPTIONS

In-house sustainment assumes the receiving team includes developers with at least junior-level experience in interactive applications, such as Unity or Unreal. ForgeFX provides onboarding and documentation to support this transition; however, baseline familiarity with the underlying technology is expected.

2.08 PERFORMANCE & SCALABILITY

The solution shall support multi-user classroom environments

CLAIM

Yes. ForgeFX trainers support both individual use and classroom-based deployment, with the ability to scale to multiple concurrent learners.

For classroom environments, instructor tools support monitoring and administration of approximately 8–24 learners per session, with larger cohorts supported through asynchronous delivery. Each learner runs an independent session, allowing groups to scale without shared runtime constraints. For scenarios requiring shared virtual interaction, the platform supports small-group co-presence, typically up to 4–8 users depending on simulation complexity. Performance targets are defined per deployment modality (e.g., XR frame rate, desktop GPU usage) and validated through automated testing to ensure consistency across releases.

PROOF

StrataTech's OcuWeld VR welding trainer demonstrates this approach. It is deployed as a WebGL application and used in trade-school environments through multi-week training programs. Student authentication is managed through Canvas LMS, and each learner progresses independently while working through the curriculum. Cohort progress is synchronized back to the LMS, allowing instructors to track performance without requiring a separate ForgeFX-hosted reporting system.

TECHNICAL DETAILS

Performance targets are defined by deployment modality, including XR frame rate requirements, desktop GPU utilization, and network usage for multi-user scenarios. Representative training scenarios are continuously tested against these targets and builds that do not meet performance thresholds are flagged during the testing process. Classroom deployments typically operate on local networks, allowing multiple instructor-led sessions to run in parallel within a single environment. Where applicable, hosted deployments can scale to support additional users through standard web or cloud infrastructure.

ASSUMPTIONS

Achievable concurrency levels depend on simulation complexity, hardware performance, and network conditions. For shared virtual sessions (co-presence), performance is influenced by the number of synchronized elements and interaction complexity. For classroom deployments, overall capacity is primarily driven by individual device capabilities and the configuration of the local or hosted environment.

2.09 USER & OPERATIONAL REQUIREMENTS

The solution shall provide instructor tools for:

- Monitoring learner progress
- Managing scenarios
- Conducting debrief and review

CLAIM

Yes. The ForgeFX platform provides integrated instructor tools for real-time learner monitoring, scenario configuration, and after-action review. Class management features such as student rostering, scenario assignment, and results export are part of the core workflow rather than separate applications. This supports consistent operation across a range of deployment sizes, from small groups to larger training cohorts.

PROOF

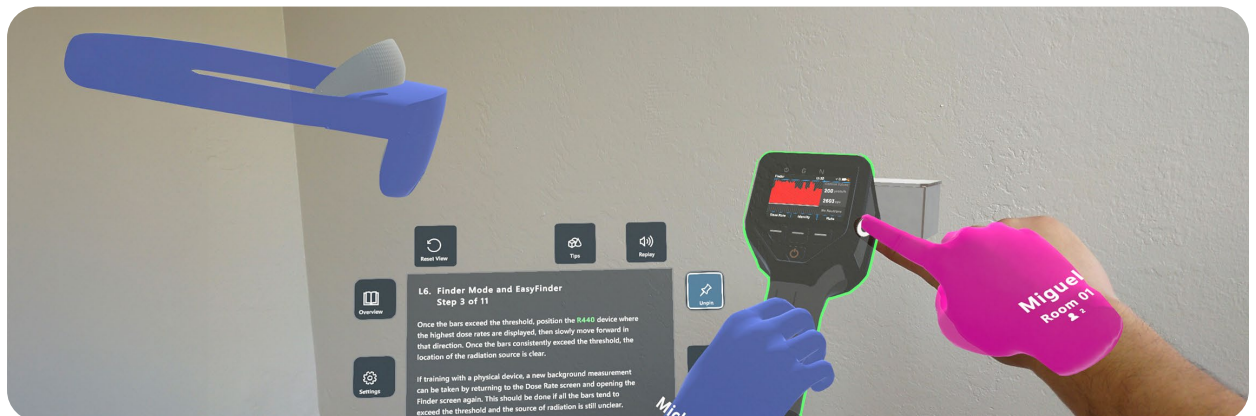
The **Global Ground Support** (GGS) configurable-scenario trainer developed by ForgeFX demonstrates this capability. Instructors configure training scenarios prior to a session, including parameters, environmental conditions, and event sequencing, and then review learner performance after completion. The system supports both pre-session setup and post-session debrief, reflecting a typical instructor-led training workflow in an operational environment.

TECHNICAL DETAILS

Real-time monitoring provides visibility into learner position, action history, error counts, and time-on-task. Scenario management tools allow instructors to start, pause, branch, inject faults, and reposition learners during execution. Session data is recorded for playback, annotation, and after-action review, enabling debrief based on the actual performance record. Reporting outputs can be exported using industry-standard formats such as SCORM, AICC, and xAPI for integration with a customer's Learning Management System (LMS).

ASSUMPTIONS

The specific metrics and data points required for debriefing and reporting should be defined in collaboration with subject-matter experts early in the project to ensure alignment with learning objectives. Integration with a customer's LMS is assumed to be part of the project scope where required.



2.10 LEARNER EXPERIENCE

The solution should:

- Provide intuitive and responsive training experience
- Support a range of interaction modalities depending on learning objectives

CLAIM

Yes. The learning experience is designed to be intuitive and responsive, with interaction methods tailored to each modality. While the core training logic and content remain consistent, input methods adapt to the target device (e.g., hand gestures in VR, touch controls on tablets).

The platform supports multiple training modes to meet different learning objectives, including Guided (with prompts), Unguided (practice), and Assessment (timed and scored). Performance metrics are reported to the instructor and can be exported to an LMS. Accessibility features, such as VR comfort settings, are included.

PROOF

The JLG equipment trainer demonstrates this approach. It provides training for boom lifts, scissor lifts, and telehandlers across multiple modalities, including an immersive VR experience with real-world hardware controls, a desktop version for classroom use, and a touch-enabled tablet version. The same training content is delivered across these formats, with interaction methods adapted to each platform.

TECHNICAL DETAILS

Training content is structured to support multiple delivery formats aligned with different learning objectives. A single procedure can be presented as a desktop walkthrough, a touch-based drill, or an immersive VR experience, depending on the training context. Core application logic manages step sequencing, completion tracking, and performance scoring consistently across modalities, ensuring a unified record of learner progress. Accessibility is incorporated into the design, including VR comfort options such as teleport locomotion and vignette modes. Error handling is designed to support recovery and continued learning rather than forcing a full restart.

ASSUMPTIONS

The choice of modality for any training activity is driven by the learning objective rather than platform constraints. The same core content may be delivered as a desktop walkthrough, touch-based drill, or immersive VR simulation, depending on the training need.



IN CONCLUSION

ForgeFX brings over two decades of experience delivering immersive training systems for complex, high-consequence operations. Our modular ForgeSIM™ framework reduces program risk, accelerates development timelines, and supports long-term sustainment without requiring re-architecture as requirements evolve.

This approach enables training ecosystems that grow over time, incorporating new platforms, procedures, and technologies while maintaining consistency and performance.

We welcome the opportunity to engage further, align on your specific objectives, and explore how ForgeFX can support the development of a robust, future-ready Virtual Maintenance Training capability.