FDE Field Delivery: Bringing AI Employees into Real Workflows with Transferable Assets

Turn workflows, knowledge, tools, permissions and acceptance into reusable delivery assets, starting at the business site

Online technical whitepaper

6 min read

Published / updated
2026-11-22

Intended readers

Enterprise project owners, business leaders, IT and security teams, delivery managers and procurement professionals

EXECUTIVE ANSWER / DIRECT ANSWER

FDE turns general AI capability into a work system that can be accepted in the customer’ s real environment. Field work is neither permanent substitution for the customer team nor unlimited customization. It uses joint discovery, short prototype cycles, real-data integration, staged acceptance and asset handover to close the gap between a demonstration and usable work.

01 / KEY QUESTIONS

Key questions answered in this paper

Q1

How is FDE different from conventional software implementation?

FDE handles workflow, model behavior, knowledge, tools and permissions together, calibrating against real tasks rather than only installing and configuring predetermined functions.

Q2

Does field delivery imply unlimited development?

No. Scope, change control, acceptance and exit criteria must be agreed, with reusable assets favored over unmaintainable personal scripts.

Q3

What should remain after delivery?

Versioned applications, interface and permission inventories, evaluation sets, operating manuals, training records, known limitations and rollback—not just a running environment.

02 / Full paper

01

Why AI employees need field engineering

The hard problems usually sit in tacit workflow, imperfect data, system permissions and accountability—not one model call.

A generic demo assumes clean inputs, explicit goals, available APIs and someone checking results. Real sites contain non-standard documents, conflicting definitions, legacy systems, account privileges, approval chains and exceptions. FDE brings business and engineering together to turn these tacit conditions into testable task contracts.

CAICT’s AI Native research emphasizes coordination among organization, process, data and technology. FDE can execute that coordination, but it cannot replace management decisions about process or automatically resolve every legacy-system problem.

- ✓ Business teams provide real tasks, exceptions and acceptance standards.
- ✓ Engineering handles runtime, knowledge, interfaces, permissions and observability.
- ✓ Security and compliance join during design rather than immediately before launch.
- ✓ The project owner remains accountable for scope, resources, change and adoption.

Chapter sources [1] [2] [4]

02

Field discovery: find the first task chain worth delivering

Priority depends on value, feasibility, risk and reuse—not only on what stakeholders most want.

Observe how work actually happens, following tasks from initiation to acceptance and collecting examples, time, roles, systems, rework and exceptions. Do not rely only on an idealized interview process. Use authorized minimal samples for sensitive data rather than copying unrelated material for research.

Score candidate tasks on business frequency, human baseline, input availability, output inspectability, action reversibility, integration cost and reuse. Scores aid decisions; they are not fabricated ROI commitments. Prefer work that can produce closed-loop evidence within one delivery cycle.

Discovery dimension	Field question	Delivery decision
Task value	Where do frequency, waiting and rework occur?	Whether to prioritize a pilot
Data & knowledge	Are source, owner, quality and permission clear?	Whether access is lawful and stable
Systems & actions	Are APIs available and errors reversible?	Automation depth and human checkpoints
Acceptance & reuse	Who accepts and can another team reuse it?	Scope and expansion path

Chapter sources [1] [4] [3]

03

Prototype co-creation and integration: close a small loop first

A prototype is not an interface demo; it verifies end-to-end work under real constraints.

Create a task contract with a small but real evaluation set, then connect essential knowledge and read-only tools before gradually enabling side-effecting actions. Change a limited number of variables per increment so quality or latency shifts can be attributed. Pin model, application, Skill, knowledge and interface versions to avoid irreproducible behavior.

VROS Agent can organize applications, events, memory, spaces and Worker execution; AIBotStore provides role and task configuration. FDE maps these capabilities to enterprise accounts, knowledge sources, business systems, approvals and compute. VR Lang, Lianshan and connector availability depend on the delivered version and project adaptation.

- ✓ Validate knowledge and retrieval read-only before writes or external actions.

- ✓ Define identity, fields, timeout, retry, rate limit and error semantics for every interface.
- ✓ Treat web pages, documents and tool output as untrusted input.
- ✓ Use simulation, approval and rollback for critical actions instead of experimenting in production.

Chapter sources [1] [8] [2]

04

Joint testing and acceptance: test quality, exceptions and responsibility

Acceptance proves repeatability, stopping and recovery under agreed conditions—not one success.

Cover normal tasks, boundary cases, missing data, insufficient permission, interface timeouts, model failures and human intervention. Record results by task stage. Business teams accept quality, IT and security review permissions and logs, and performance is measured in the actual model-serving and compute environment. State any differences when masked or simulated data is used.

Acceptance conclusions need thresholds, sample scope, versions, owners and raw evidence. Target productivity, delivery-cycle reductions, refusal rates or hardware figures that were not validated cannot become project facts. Failed items require explicit restriction, remediation, retest or termination.

Acceptance layer	Test content	Sign-off evidence
Business quality	Real tasks, difficult cases and delivery standards	Evaluation results and business sign-off
Technical operation	Repetition, timeout, recovery and sustained running	Events, logs and resource records
Security & permissions	Privilege escalation, injection, approval and data boundaries	Security tests and remediation
Organizational operation	Training, takeover, shutdown and escalation	Exercise records and owner confirmation

Chapter sources [6] [5] [2]

05

Asset handover: enable the customer to continue operations

Delivery ends when the enterprise can understand, maintain and audit the work assets—not when an expert leaves.

The handover inventory includes VR Lang or other application assets, prompts and Skills, knowledge and interface inventories, permission matrices, evaluation sets, monitoring, operating manuals, training and known limitations. Every asset needs an owner, version, dependencies, storage location and change method; a script on one laptop is not a handover.

ISO/IEC 42001’s management-system approach can inform responsibility, risk, operations and improvement, but referencing it does not mean AIBotStore or the customer project is certified. Asset ownership, third-party licenses, confidentiality, exit and ongoing support should be separately defined in contract and handover records.

- ✓ Applications: version, purpose, dependencies, release and rollback.
- ✓ Data: source, permissions, quality, update and deletion responsibility.
- ✓ Evaluation: samples, baselines, thresholds, failures and retest method.
- ✓ Operations: monitoring, alerts, takeover, training and change procedures.

Chapter sources [7] [4] [6]

06

Change and ongoing service: prevent field dependency

FDE should increase customer operating autonomy rather than create an irreplaceable human black box.

Use joint reviews and a change board to assess new requests by value, risk and dependencies. Changes to models, tools, knowledge, permissions or processes trigger impact analysis and necessary retesting. Recurring common problems should become productized configuration, standard connectors or shared playbooks rather than repeated bespoke work.

Post-deployment monitoring must connect with customer support, security and asset management. Controlled self-evolution, where included, produces candidate experience and improvements only, followed by sandbox validation, human approval, versioning and controlled release. Availability depends on delivery and acceptance;

accountable humans remain responsible for high-impact actions.

- ✓ Define service boundaries, response paths, change classes and commercial principles.
- ✓ Turn repeated issues into documentation, tests and reusable components.
- ✓ Use staged acceptance to expand, maintain, reduce or exit.
- ✓ Continuously review quality, security, cost and human takeover capability.

Chapter sources [6] [7] [1]

03 / CHECKLIST

Pre-project checklist

Use these items to structure a PoC plan, procurement questions or acceptance checklist.

01	Observe real work and collect authorized representative samples
02	Select the first task chain by value, feasibility, risk and reuse
03	Move from read-only prototypes to tools and side effects incrementally
04	Accept quality, repetition, exceptions, security and human takeover together
05	Hand over versioned applications, interfaces, permissions, evaluations and manuals
06	Establish change, retest, support, exit and customer operating autonomy

04 / FAQ

Frequently asked questions, answered directly

01

Must FDE remain on site permanently?

Not necessarily. Field intensity depends on discovery, integration and acceptance, and can combine on-site and remote work. The key is authentic field information, clear responsibility and transferable assets.

02

How is FDE distinguished from ordinary custom development?

Define task chains, model and knowledge behavior, reusable assets and acceptance at initiation. Conventional feature development may be included, but FDE also verifies AI task quality, exceptions and human accountability.

03

Can delivery-cycle reduction or productivity multiples be promised before a PoC?

Targets should not be stated as facts. Cycle time and productivity require a customer baseline, defined scope and real acceptance, and depend on data, interfaces, process and organization.

04

Can the customer maintain the system independently afterward?

That should be a handover objective through versioned assets, documentation, training, exercises and permission transfer. The actual scope depends on contract and team capability.

05 / REFERENCES

References and public sources

Chinese references follow GB/T 7714 style and English references follow APA style. Accessed November 22, 2026. External material informs industry and technical methods; it does not imply endorsement or certification of AIBotStore.

[1] China Academy of Information and Communications Technology. (2026). AI Native research report 1.0: Enterprise transformation framework and practice [Report].

https://www.caict.ac.cn/kxyj/qwfb/ztbg/202610/P020261022558735347833.pdf →

[2] China Academy of Information and Communications Technology, & China University of Political Science and Law. (2026). Agent governance research report (2026) [Report].
https://www.caict.ac.cn/kxyj/qwfb/ztbg/202609/P020260916574260931247.pdf →

[3] China Academy of Information and Communications Technology. (2026). Artificial intelligence security governance blue book [Report].
https://www.caict.ac.cn/kxyj/qwfb/bps/202601/P020260109784447548497.pdf →

[4] National Institute of Standards and Technology. (2023). AI Risk Management Framework (AI RMF 1.0).
https://www.nist.gov/itl/ai-risk-management-framework →

[5] National Institute of Standards and Technology. (2024). Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile (NIST AI 600-1).
https://doi.org/10.6028/NIST.AI.600-1 →

[6] Rao, A., Keller, A., Kalra, N., Steed, R., Kwegyir-Aggrey, K., Klyman, K., Staheli, D., & Bergman, A. (2026). Challenges to the monitoring of deployed AI systems (NIST AI 800-4).
https://doi.org/10.6028/NIST.AI.800-4 →

[7] International Organization for Standardization. (2023). ISO/IEC 42001:2023—Information technology—Artificial intelligence—Management system.
https://www.iso.org/standard/42001 →

[8] OWASP GenAI Security Project. (2025). Agentic AI—Threats and mitigations.
https://genai.owasp.org/resource/agentic-ai-threats-and-mitigations/ →

Scope statement

This whitepaper supports technology selection and project planning and does not replace legal, compliance, safety or investment advice. Actual VROS Agent, AIBotStore, hardware-integration and controlled-evolution scope depends on the delivered version, project contract and acceptance. Industry task examples do not represent delivered customers or achieved outcomes.