SagaAIoT™

Enterprise-Grade Governed AIoT Platform

Business Case White Paper

Version 7.1 · June 2026 · Prasaga Foundation · Confidential

CONFIDENTIAL: Prepared for C-suite executives, board members, and strategic partners evaluating the SagaAIoT™ platform. Technical implementation details are withheld in accordance with enterprise distribution policy.

Executive Summary

The SagaAIoT™ platform addresses one of the most consequential gaps in enterprise IoT strategy: the absence of governed, auditable control over AI agent actuation on physical systems. As artificial intelligence transitions from advisory to operational roles — controlling HVAC, traffic infrastructure, manufacturing equipment, grid assets, and medical facilities — enterprises face compounding liability and compliance exposure that point solutions cannot address.

Prasaga Foundation has built and shipped a production-grade platform that closes this gap through three integrated components:

- **SagaIoT** — enterprise **multi-protocol message platform**: **XMPP** default full bus (**122k/s** routed @ 1 KB and **200k+** concurrent device connections per node — G1) plus **MQTT** first-class peer (**~326k/s** server publish accept @ 1 KB). Mandatory TLS, rate limiting, SCRAM admin, federation, WebSocket, clustering — shipped June 2026.
- **SagaChain** — blockchain governance and audit ledger providing policy-gated command authorization, human cosign workflows, parameter bounds enforcement, and tamper-evident forensic records — without placing IoT payload data on-ledger.
- **SagaAI** — AI agent **access-control and guard-certificate layer**: hybrid **OAuth** (delegated token/scopes), **RBAC** (roles/permissions), **ABAC** (context attributes), and **PBAC** (policy rules with permit/deny/human-approval outcomes), plus executable governance control-plane checks — positioned for EU AI Act, NIST AI RMF, HIPAA, NERC-CIP, and ISO 27001 deployments.

The platform's architecture deliberately separates speed from trust: device telemetry flows over the high-throughput message bus while material commands are policy-checked, co-signed if required, and permanently recorded on chain. No synchronous ledger write appears on the device messaging hot path.

> Phases 0 through 13 — encompassing the full XMPP message layer, MQTT transport (M0–M6, B1, B2) with device attestation parity, security hardening, multi-node clustering, cross-domain federation, WebSocket connectivity, the governed command path, and the Agent Runtime Adapter (ARA) with OAuth/RBAC/ABAC/PBAC hybrid evaluation (Phase 4 fixture + Phase 4b registry client) — are **shipped code**, verified by a **layered internal security program** (June 2026): **P0 release gate 18**

> **pass / 0 fail** (re-verified uninterrupted 27 June 2026); P1–P4 in-house adversarial phases and a 13/13 read-only production audit on sagaiot.net. This is **not** a third-party penetration test.

The remaining engineering work is **registry-mode live E2E** on a target chain (SagaAI + IoT domain load and bookmarked LOIDs), expanded SagaChain audit anchoring, and Phase 14 Blockchain Listener — improvements that strengthen an already-operational platform rather than prerequisites for initial deployment or fixture-mode pilots.

1. The Market Problem: Ungoverned AI Actuation

Connected infrastructure is now operated increasingly by software agents rather than by people at a console. When an AI agent adjusts a building thermostat, opens a valve in a water treatment facility, reroutes traffic, or modifies a power grid setpoint, the enterprise must answer four questions: who authorized the action, under what policy, within what parameter bounds, and what was the verified outcome? Today, most enterprises cannot answer any of these questions with the evidentiary quality that regulators and insurers will increasingly demand.

1.1 The Trust Gap in Enterprise IoT

The IoT messaging market is mature and well-served for transport. What it does not deliver is governed agent actuation. When an LLM-based agent, an autonomous control system, or a multi-agent orchestration framework issues a device command, there is no shared, non-disputable, auditable record between the agent, the operator, the device, and the regulator.

The EU AI Act imposes enforceable requirements on AI systems operating in high-risk categories — including infrastructure management, industrial equipment, and medical devices — with compliance obligations that are already active for many enterprise use cases. NIST AI RMF, NERC-CIP, HIPAA, and ISO 27001 frameworks are evolving in parallel.

1.2 Why Point Solutions Fall Short

Incumbent Stack	Transport Strength	Governance Gap
MQTT Brokers	High throughput, lightweight pub/sub, persistent sessions	No native policy object; no chain-backed audit; no AI agent authorization layer
Hyperscaler IoT Hubs	Managed scale, global availability, device SDK ecosystem	Platform lock-in; governed agent path is vendor-proprietary; sovereignty constraints
Purpose-Built Industrial Platforms	Domain-specific integrations, OT protocol support	Proprietary governance; limited AI agent interoperability; high switching cost
Blockchain-Only IoT	Immutable record, decentralized trust	Fundamental throughput mismatch for telemetry volume; payload latency unacceptable

2. Platform Architecture: Two Planes, One Governed Path

SagaAIoT™ is built on a deliberate two-plane architecture that separates the requirements of volume and speed from the requirements of trust and auditability. These planes are connected by a thin bridge that imposes no latency on the device messaging hot path.

Plane	Component — Role
Data Plane	SagaIoT `sagaiotd` — XMPP c2s, PubSub, MUC, presence, roster, federation, WebSocket. **Or** MQTT on customer broker / Mode B sidecar / B1 in-process. All IoT payload data stays on this plane.
Trust Plane	SagaChain — Governance classes, command lifecycle, AI policy, persistent audit state on ledger. Receives only metadata and state transitions — never device payload bodies.
Bridge	Command Bridge Sidecar — Watches chain-approved commands, publishes to **XMPP PubSub or MQTT**, awaits device acknowledgment, posts execution results back to chain. Decoupled from the message hot path.

> **Hard Rule:** No synchronous ledger write appears on the SagaIoT message routing or PubSub publish path. SagaChain receives batched metadata via sidecars only. This design principle is non-negotiable and is enforced architecturally, not by convention.

2.1 SagaIoT: The Messaging Foundation

SagaIoT provides the **data plane** — moving device telemetry and command payloads at IoT scale. **XMPP** is the default full-feature bus. **MQTT** is a first-class peer for estates that already run Mosquitto, EMQX, or cloud IoT hubs — governed commands and audit attach via the command bridge without mandating XMPP migration.

Transport	Best for	Shipped
XMPP	Greenfield, roster/MUC/federation, WebSocket clients	Phases 0–13
MQTT Mode A	Existing external broker — bridge-only, no sagaiotd XMPP	M0–M6
MQTT B1/B2	Embedded or sidecar broker inside Compose / sagaiotd	B1 in-process; B2 Mosquitto sidecar

2.2 SagaChain: The Trust and Governance Layer

SagaChain provides the immutable governance record that transforms AI agent actuation from an unverifiable event into a policy-compliant, auditable transaction.

Governance Object	Business Function
ClassIoTDevice	Registers device capability manifests including allowable command verbs, parameter bounds, and capability declarations.

Governance Object	Business Function
	Commands targeting verbs outside the manifest are rejected before dispatch.
ClassIoTDeviceCommand	The authoritative command record. Created by an agent, subjected to policy evaluation, and stamped with approved, pending cosign, or rejected status.
ClassIoTAgentPolicy	Executable policy object. Evaluates commands against device class allowlists, numeric parameter constraints, human cosign requirements, and policy revocation status.
Human Cosign Queue	Safety-critical commands requiring a second approver are held in pending cosign status. The bridge will not dispatch until cosign is received.
ClassIoTCommandResult	Closes the governance loop. Posted after device acknowledgment, providing the forensic record of what was dispatched, what the device confirmed, and when.
ClassDeviceComplianceBinding	Links device-command records to applicable regulatory profiles, enabling compliance teams to generate evidence packages per framework without manual log correlation.

2.3 SagaAI: Four-Layer Access Control for AI Agents

Before an AI agent's command enters the IoT governed path, SagaAI evaluates **four deterministic layers** — no probabilistic model judgment in the gate:

Layer	Business question	What it enforces	Example (HVAC actuation)
OAuth	Did this agent receive a valid delegated grant?	Token validity, expiry, revocation, scopes	Agent presents scope command:actuate issued to a facilities automation service account
RBAC	Does this identity hold a permitted role?	Roles and permissions — least-privilege mapping	Session includes role OT_Operator with permission command:actuate on ClassHVACZone
ABAC	Given current context, should this identity act now?	Attributes: risk score, time window, location zone, device attestation state, human approval flag, playbook step	risk_level=low, time_window=daytime, location_zone=zone-a must match policy context
PBAC	Which organizational policy applies, and what is the outcome?	Policy rules with effects: permit, deny, or require human approval; obligations	Policy "HVAC actuation daytime low risk" permits actuation only when ABAC attributes and

Layer	Business question	What it enforces	Example (HVAC actuation)
			OAuth/RBAC are satisfied

Evaluation produces an immutable **access decision** and, when permitted, a **guard certificate** from the executable governance control plane. The **Agent Runtime Adapter (ARA)** is the production sidecar running this stack for AI tool APIs: **fixture mode** for pilots without a live chain; **registry mode** submits the same logic as on-chain transactions once governance domains are loaded.

2.4 The Governed Command Path

The command lifecycle is deterministic and non-bypassable by design:

1. An AI agent (via **ARA**) or human operator passes **OAuth** → **RBAC** → **ABAC/PBAC** evaluation and receives a guard certificate when permitted.
2. A **ClassIoTDeviceCommand** is created on SagaChain (registry mode) or approved in fixture configuration.
3. **ClassIoTAgentPolicy** evaluates the command: device class and verb must match allowlists; numeric parameters must satisfy configured bounds; cosign requirements are applied if the verb is classified as safety-critical.
4. The command bridge polls SagaChain for bridge-eligible commands — those in approved or dispatched status only. Rejected and pending-cosign commands are not dispatched.
5. Eligible commands are published to the device transport — **XMPP PubSub node or MQTT topic** per chain `transport_profile`.
6. The target device publishes an acknowledgment to the designated results node or topic.
7. The bridge posts a ClassIoTCommandResult to SagaChain, closing the audit record.

> **Key Differentiator:** The bridge is the only dispatch path for governed commands. There is no shadow IT route through which an agent can bypass policy evaluation and issue a direct device command. Devices subscribe only to known PubSub nodes registered on chain.

3. Platform Capabilities — Current Shipped State

3.1 Cross-Domain Federation

SagaIoT Phase 13 Track A delivers XMPP server-to-server federation with a statically peered allowlist trust model. Shipped capabilities include bidirectional s2s connections (XEP-0288), Dialback authentication (XEP-0220), optional TLS on s2s, optional SRV DNS discovery, MUC relay across federated domains, PubSub relay across federated domains, multiple XMPP domains in a single sagaiotd process, per-peer feature caps, and federation metrics in the Admin UI.

3.2 WebSocket Client Connectivity

SagaIoT Phase 13 Track B ships RFC 7395 XMPP-over-WebSocket, enabling browser-native clients with the same authentication, session management, and routing as TCP clients. Operator deployment uses Caddy reverse proxy to terminate WSS connections.

3.3 Horizontal Multi-Node Clustering

SagaIoT Cluster Track (Phases 1–6) delivers horizontal scaling across multiple sagaiotd pods. The cluster bus supports NATS with JetStream and Redis backends. Cross-node stanza delivery and PubSub cross-node fan-out are validated by the cluster smoke test suite. Clustering is opt-in.

3.4 Security Hardening (Phase 12)

- **Mandatory TLS** on all client-to-server connections — plaintext connections are rejected
- **SCRAM-authenticated administration** — admin API requires SCRAM login yielding a Bearer token session
- **Connection and message rate limiting** — configurable per-connection limits protect against denial-of-service
- **Audit sinks** — operator-configurable file and HTTP event sinks for SIEM integration
- **Async audit architecture** — audit event emission is off the message hot path

3.5 Agent Runtime Adapter and SagaAI Access-Control Stack (Phase 4 + 4b)

Capability	Description	Status / Validation
Agent Runtime Adapter (ARA)	HTTP sidecar for AI agent tools — iot_evaluate_policy, iot_command, iot_observe — before commands become bridge-eligible	**Shipped. verify-ara-no-chain.sh; mocked registry client tests**
OAuth + RBAC + ABAC + PBAC	Four-layer hybrid authorization; immutable access decision + guard certificate	On-chain class trees + deterministic unit tests; ARA fixture engine
Registry client (Phase 4b)	Submits SagaIoT_ara_* txn templates via node HTTP API	**Shipped. Mocked CI; live E2E pending domain load on target chain**

Fixture mode enables full hybrid-governance demonstrations without chain registration. Registry mode is the production path once SagaAI and IoT governance domains are loaded.

4. Competitive Positioning

Category	Representative Stacks	Gap vs. SagaAIoT™
MQTT Brokers	Eclipse Mosquitto, HiveMQ, EMQX, VerneMQ	Transport only — no chain policy objects, no governed agent lifecycle. **SagaAIoT adds governance via bridge on the customer's existing broker (Mode A)** without replacing it.
Hyperscaler IoT Hubs	AWS IoT Core, Azure IoT Hub, Google Cloud IoT	Managed convenience at the cost of lock-in; governed agent path is platform-proprietary; sovereign and

Category	Representative Stacks	Gap vs. SagaAIoT™
		regulated-sector deployments face data residency constraints; no portable governance layer.
Purpose-Built Industrial Platforms	ThingWorx, Losant, XMPro	Strong domain-specific integrations; governance models are proprietary; AI governance is an add-on, not a foundational architecture.
Legacy Proprietary XMPP Platforms	Incumbent enterprise XMPP deployments	High operational cost; no chain integration; no development path for four-layer AI agent governance.

SagaAIoT™ is the only architecture in this landscape that combines a **multi-protocol data plane** (XMPP full bus or MQTT on existing broker estates), portable self-hosted blockchain governance (SagaChain XBOM) not tied to a hyperscaler, an **executable governance model** with policy objects that run code, **off-ledger device telemetry with on-ledger command governance**, and **sovereignty-compatible deployment**.

5. Enterprise Use Cases

5.1 Smart City Infrastructure

Municipal AI systems managing traffic signals, streetlighting, water distribution, and public transit require attributable, policy-compliant, auditable decisions for every infrastructure change. SagaAIoT™ enables federation across municipal agencies while maintaining per-agency domain sovereignty. The governed command path ensures AI-initiated infrastructure changes are policy-checked, bounded to safe parameter ranges, and recorded permanently before any physical actuation occurs.

5.2 Industrial Operations

Manufacturing, energy, and process industries operate physical systems where an out-of-bounds command can cause equipment damage, production loss, or safety incidents. The platform's parameter constraint enforcement in policy objects is non-bypassable: an AI agent cannot issue a temperature setpoint outside the configured maximum, regardless of what the model computes. The constraint is in the governance layer, evaluated before the command reaches the bridge.

5.3 Healthcare-Adjacent Environments

Facility management AI in healthcare settings — HVAC, access control, medical gas systems — operates under HIPAA facility safeguards. SagaChain's compliance binding objects enable operators to link device command records directly to applicable regulatory profiles, generating evidence packages that reduce audit preparation time from weeks to hours.

5.4 Energy and Grid Operations

NERC-CIP requirements for bulk electric system cyber security impose strict controls on who may command grid assets. SagaAIoT™'s combination of mandatory authenticated sessions, TLS transport security, and chain-backed command authorization provides a defensible compliance architecture. The

immutable command record supports NERC-CIP audit evidence requirements without manual log reconstruction.

6. Business Value and Return on Investment

Value Driver	Mechanism and Enterprise Impact
Operational Risk Reduction	Rejected commands never reach devices. Parameter bounds are enforced before dispatch. Human cosign requirements prevent autonomous agent action on safety-critical verbs. ABAC/PBAC context-aware escalation triggers human oversight dynamically when operational risk exceeds configured thresholds.
Audit and Incident Response Cost	Chain objects replace manual log correlation. Compliance binding objects link command records to regulatory framework evidence requirements. Incident response time for AI-actuated events is reduced from days of log reconstruction to on-demand chain query.
Platform Independence	Open XMPP and MQTT standards plus self-hosted stack eliminates hyperscaler lock-in. MQTT-first estates keep their broker; governance is additive via bridge. The SagaChain XBOM layer is a portable artifact, not a proprietary vendor dependency.
AI Deployment Enablement	Enterprises that cannot demonstrate provable AI governance controls face growing barriers to deploying AI agents in regulated contexts. SagaAIoT™ provides the technical foundation for EU AI Act compliance, NIST AI RMF alignment, and procurement positioning in sectors requiring demonstrated AI governance maturity.
Incremental Adoption Economics	The governed command path is an additive sidecar to the SagaIoT message bus. Enterprises can deploy messaging first, validate operations, and activate governed command workflows when AI agent deployment or compliance requirements demand it.

7. Security Architecture and Compliance Posture

7.1 Transport and Session Security

All client-to-server connections require TLS — plaintext connections are rejected at the protocol level. MQTT devices use the same provisioned X.509 trust anchor as XMPP (factory or site CA). Server-to-server federation connections support TLS with a shared CA for mutual peer verification; TLS-verified peers bypass Dialback.

7.2 Governance-Layer Authorization

Command authorization operates at two complementary levels:

- **SagaAI hybrid access control (AI agents):** OAuth delegated scopes, RBAC roles, ABAC context attributes, and PBAC policy rules are evaluated in sequence before a guard certificate is issued. Failure at any layer blocks the command and records an access decision suitable for audit.

- **IoT agent policy (all commands):** Executable policy on SagaChain evaluates device class, verb allowlists, parameter bounds, and human cosign requirements before bridge dispatch.

Neither layer is a middleware check or application permission table. Policy objects on the ledger must reach consensus (registry mode) or are enforced deterministically in fixture mode before dispatch is eligible. This is materially stronger than access control lists in application code, which can be bypassed by defects, misconfiguration, or compromised service accounts.

7.3 Regulatory Framework Alignment

Framework	SagaAIoT™ Alignment
EU AI Act	High-risk AI system controls — provable policy enforcement, audit trail, human oversight capability (Art. 14), and parameter bounds enforcement. ABAC/PBAC dynamic escalation directly maps to Art. 14 human oversight requirements for high-risk actuation.
NIST AI RMF	Govern, Map, Measure, Manage functions supported by executable governance objects (Govern), device registry and capability manifests (Map), command result records and ack tracking (Measure), and human cosign and policy revocation (Manage). NIST SP 800-162 ABAC model implemented in ClassABACAttribute and ClassPBACPolicy.
HIPAA Facility Safeguards	Audit controls, access management, and integrity controls for facility AI management systems addressed through authenticated sessions, chain-backed command records, and compliance binding objects.
NERC-CIP	Cyber security controls for bulk electric system assets supported through mandatory authenticated sessions, rate limiting, TLS transport, and immutable command records with policy attribution.
ISO 27001 / ISO 42001	Information security and AI governance management alignment through layered access controls (OAuth/RBAC/ABAC/PBAC), audit event logging, secure transport, and governance-layer authorization for system actuation.

7.4 Internal Security Verification (June 2026)

SagaAIoT™ security claims for pilots and RFPs should reference internal engineering evidence, not imply third-party penetration testing.

Capability	Description	Status / Validation
P0 — Release gate	make platform-vuln-bundle: CVE/SAST/greps, MQTT/XMPP smokes, governed lab, ARA fixture verify	**18/18 pass — re-verified 27 June 2026**
P1 — Adversarial phase 1	make adversarial-phase1: Authz matrix, topic traversal	**3/3 pass**
P2 — Adversarial phase 2	make adversarial-phase2: Fuzz, bridge traversal, ack spoof	**3/3 pass**

Capability	Description	Status / Validation
P3 (lab) — Adversarial phase 3	make adversarial-phase3: Bounded abuse, CVE scan	**4/4 pass**
P3 (prod) — Read-only audit	Audit of sagaiot.net live reference deployment	**13/13 pass**
P4 — Federation/chain	make adversarial-phase4: Federation matrix, chain preflight	**3/3 pass**
External pen test	Third-party penetration test	Not done — required before strongest external security statements

8. Deployment Model and Pilot Approach

8.1 Deployment Configurations

Configuration	Purpose	Components
Single-Node	Evaluation, development, small-scale pilot	sagaiotd + Postgres + Admin UI. Default compose stack.
Multi-Node Cluster	Production scale, horizontal redundancy	Two or more sagaiotd pods + NATS JetStream cluster bus + shared Postgres.
Cross-Domain Federation	Multi-organizational or multi-agency deployments	Separate sagaiotd instances per domain, s2s peered, with optional MUC and PubSub federation.
Chain-Integrated	Governed command path + AI agent governance	SagaIoT stack, command bridge sidecar, ARA (OAuth/RBAC/ABAC/PBAC), SagaChain integration. Fixture mode for pre-registration demos; registry mode after governance domain load.
Production	Enterprise deployment with TLS and secrets management	Production compose template with Caddy TLS, environment-file secrets, no published database port, resource limits, and backup runbook.

8.2 Phased Adoption Path

8. **Phase 1 — Messaging Foundation:** Deploy SagaIoT single-node or cluster. Validate device connectivity, telemetry ingestion, PubSub patterns. Estimated timeline: two to four weeks to first device traffic.

9. **Phase 2 — Security and Operations Hardening:** Enable mandatory TLS, configure SCRAM admin authentication, establish rate limiting, connect audit sinks to SIEM.
10. **Phase 3 — Federation (if multi-domain required):** Configure s2s federation to peer domains. Validate cross-domain messaging, MUC, and PubSub relay.
11. **Phase 4 — Governed Command Path + ARA:** Deploy command bridge and Agent Runtime Adapter. Demonstrate OAuth/RBAC/ABAC/PBAC hybrid evaluation and IoT agent policy in fixture mode. Load governance domains on SagaChain testnet for registry-mode pilots when ready.
12. **Phase 5 — Compliance Evidence and Audit:** Configure compliance binding objects for applicable regulatory framework. Generate first audit evidence package.
13. **Phase 6 — Production Rollout and Expansion:** Promote to production. Expand device registry, agent policies, command set, and ARA registry-mode LOIDs on target chain.

9. Honest Assessment of Current Limitations

Prasaga publishes a candid accounting of platform limitations as a trust asset, not a liability.

Item	Current Status	Path Forward
Governance Domain on Chain	Governance classes and SagaAI access-control trees are shipped and unit-tested. Pilots use fixture mode (full path) or ARA registry client (txn templates); live registry E2E requires domain load on target node.	Load order: executable governance → OAuth/RBAC → ABAC/PBAC → IoT governance (load-ara-registry.sh).
ARA registry mode live E2E	Registry client + txn templates shipped; mocked CI tests pass.	Bookmark LOIDs on loaded node; set ARA_CHAIN_MODE=registry.
Device Attestation (mTLS / SASL EXTERNAL + MQTT parity)	Shipped (opt-in) — per-device X.509 on XMPP and MQTT (B1/B2); PQ Model B bind; payload attestation; B2 sidecar audit worker; audit-anchor sidecar; Merkle batch anchor.	Enable SAGAIOT_DEVICE_CA_CERT; make platform-vuln-bundle attestation smokes.
Dynamic Engagement	Shipped (opt-in) — rules, pools, Ushers, admin Engagement tab.	See scripts/seed-engagement-lab.sh.
SagaChain Audit Anchor from Message Bus	Shipped (opt-in) via integration/audit-anchor/ sidecar.	Async batches off message hot path; Blockchain Listener (Phase 14) planned.
BOSH Long-Poll c2s	Deferred. WebSocket c2s is shipped (Phase 13 Track B).	Activated by named pilot requirement. Not currently prioritized.
markFailed for Timeout / Failed Ack	Command bridge currently posts markExecuted in all cases.	Task 5 in governed command path engineering backlog.

Item	Current Status	Path Forward
	Distinct failure path for timeout or negative ack is planned.	
Production k8s / Multi-VPS Validation	Single-node and cluster docker compose are validated. Kubernetes and multi-VPS production rollout are not yet validated.	Activated by first production enterprise deployment.

10. Engineering Roadmap

Track / Phase	Scope and Objective
Phase 9 — Production VPS Verify	Validate production docker-compose.prod.yml on a Linux VPS with remote smoke bot, Caddy TLS, and nightly Postgres backup. Gate for enterprise deployment confidence.
Phase 14 — SagaChain Integration	Blockchain Listener sidecar, contract-bound session APIs, registry-mode bridge hardening, automated result POST.
Track B — Device Attestation (Gaps 1–5)	Shipped (opt-in) — mTLS/SASL EXTERNAL, payload signing, audit-anchor, Merkle batching, proof-of-delivery.
Dynamic Engagement (DE-1–5)	Shipped (opt-in) — rules/pools/Ushers, webhook + gRPC Ushers, NATS bus, admin UI.
SagaAI Guard Certificates (Phase 4 + 4b)	Shipped via ARA — hybrid guard + guard cert in fixture mode; registry client submits chain txns; live E2E pending domain load on target node.
Protocol Depth (as pilot-driven)	XEP-0077 in-band device self-registration, additional XEP coverage per enterprise pilot requirements. Activated by named customer need.

11. Next Steps

SagaAIoT™ is available for enterprise pilot engagement today. Prasaga recommends the following approach for prospective partners:

- **Executive Briefing:** A 60-minute briefing covering platform architecture, competitive differentiation, pilot design, and compliance alignment. Intended for CTO, CIO, CISO, and Risk Officer audiences.
- **Technical Pilot Design Workshop:** A half-day working session to define the pilot device class, agent policy configuration, cosign verb selection, and success metrics. Output: a written pilot scope document.
- **Pilot Deployment:** Prasaga provides deployment support for the SagaIoT messaging stack and command bridge. Target: first governed command lifecycle demonstration within 30 days of pilot scope agreement.

- **Compliance Alignment Review:** Engagement with your compliance or legal team to map platform capabilities to applicable regulatory framework requirements. Prasaga provides technical evidence documentation.
- **Partnership and Commercial Terms:** Discussion of licensing, support, and long-term partnership structure following successful pilot completion.

Appendix A — Glossary

Term	Definition
SagaAIoT™	Prasaga Foundation's integrated enterprise platform combining SagaIoT, SagaChain, and SagaAI.
SagaIoT	Multi-protocol message layer — XMPP full bus (default) and MQTT peer transport — component of SagaAIoT™.
SagaChain	Governance classes, persistent command state, AI policy, and immutable audit ledger — where platform function lives.
SagaAI	AI agent OAuth/RBAC/ABAC/PBAC access control, guard certificates, and executable governance — component of SagaAIoT™.
Agent Runtime Adapter (ARA)	Sidecar HTTP API connecting AI agent tools to hybrid access-control evaluation and governed IoT commands.
OAuth / RBAC / ABAC / PBAC	Four-layer authorization: delegated token/scopes; roles/permissions; context attributes; policy rules with permit/deny/human-approval outcomes.
Access Decision	Immutable record of hybrid authorization evaluation — outcome, context snapshot, obligations.
Guard Certificate	SagaAI object recording structural pre-execution checks after OAuth/RBAC/ABAC/PBAC and control-plane evaluation.
Command Bridge	Sidecar that watches SagaChain for approved device commands, dispatches to XMPP PubSub or MQTT, and posts execution results back to chain.
Governed Command	A device command that passed SagaChain policy evaluation, is recorded immutably, and is eligible for bridge dispatch.
XBOM	Extensible Business Object Model — SagaChain's portable governance artifact format.
Human-in-the-Loop (HITL)	Cosign requirement holding safety-critical commands pending human approval before dispatch. Also triggered dynamically by ABAC/PBAC context evaluation.
MQTT transport	Topic-based publish/subscribe (sagaiot/v1/{domain}/...); Mode A uses customer broker without sagaiotd XMPP.

Term	Definition
transport_profile	Chain field on device registration — selects XMPP, MQTT, or hybrid delivery for the bridge.
Fixture Mode	ARA or command-bridge mode using local JSON configuration instead of live SagaChain registry poll. Enables full governed path and hybrid OAuth/RBAC/ABAC/PBAC demonstration prior to chain registration.
Registry Mode	ARA or sidecar submits SagaPython txn templates via node HTTP API. Client shipped; live E2E requires domains loaded on target chain.

Version 7.1 · June 2026 · Confidential · www.prasaga.com · sagaiot.net · sagascan.prasaga.com