CONFIDENTIAL

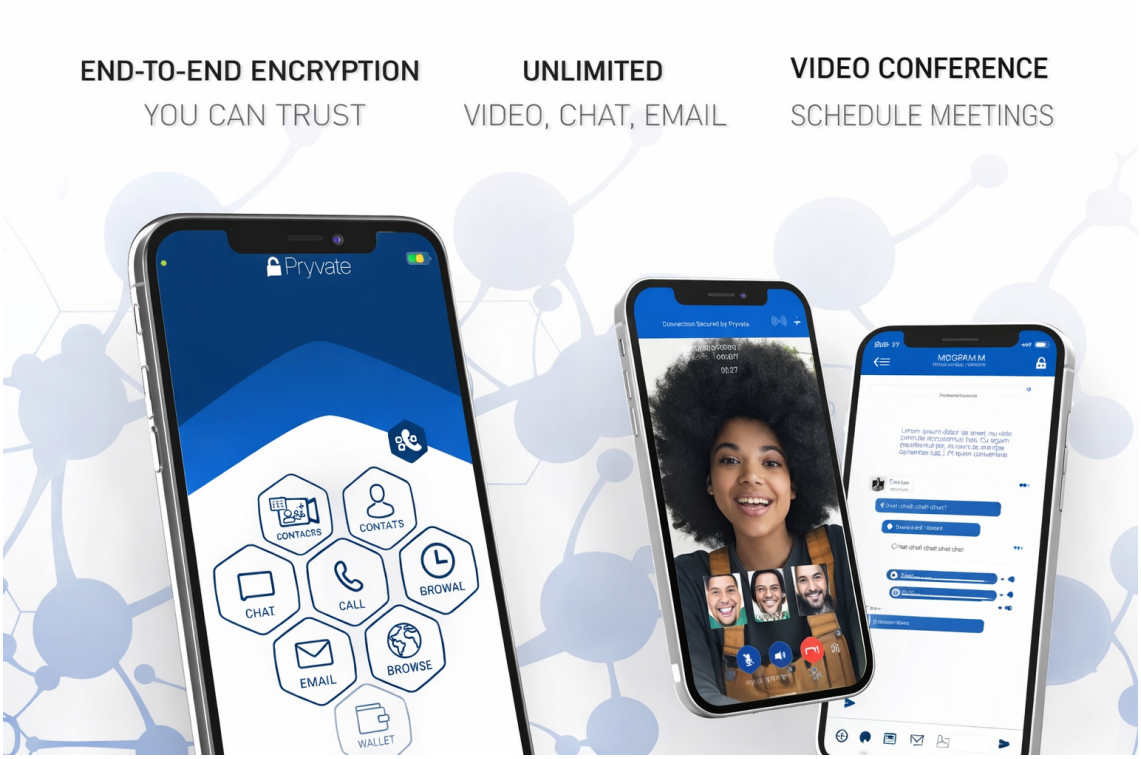
PRYVATE INTEGRATOR SDK

Secure Communications Infrastructure for Platform Integrators

Complete Board-Level Governance, Security & Compliance Dossier

Version: 1.0
Date: 28 March 2026
Prepared For:

_____
Prepared By: Pryvate



1. Introduction

This document provides a comprehensive overview of Pryvate Integrator SDK — a secure communications infrastructure layer designed for integration into third-party platforms.

It is intended for technical, architectural, and governance stakeholders who require a clear understanding of how secure messaging, voice, and video communication can be embedded within regulated or high-trust applications without introducing additional operational or compliance risk.

Pryvate Integrator is not a standalone communications product. It is not deployed as an independent user-facing application, nor does it operate as an external messaging service. Instead, it functions as an embedded communications subsystem, integrated directly into the host platform and governed by the integrator's backend systems.

In practical terms:

- The host application retains full ownership of user identity, workflow, and access control
- Communication capability is introduced through a controlled SDK layer
- All communication remains within the boundaries of the platform

This model allows platforms to introduce real-time communication without relinquishing control to external systems.

Pryvate Integrator does not introduce a separate user-facing environment and does not require users to adopt an external communication application.

Pryvate Integrator is typically deployed in platforms operating in regulated or high-trust environments, including financial services, healthcare, legal systems, and government applications.

1.1 Executive Risk Summary

For digital platforms, communication represents a material and often under-recognised risk surface.

Sensitive information — including financial instructions, healthcare data, legal discussions, and strategic decisions — is routinely transmitted through communication channels that are not designed for platform-level governance. In many cases, platforms rely on external messaging tools or ad hoc communication workflows that sit outside their control boundary.

This creates structural exposure across several dimensions:

- Loss of control over where and how communication occurs
- Inability to enforce access policies consistently
- Lack of lifecycle governance over communication participants
- Persistence of sensitive data outside platform boundaries
- Increased regulatory and operational risk

Building communication infrastructure internally introduces a different form of risk:

- Significant engineering complexity
- Long-term maintenance burden
- Security architecture challenges
- Extended delivery timelines

Pryvate Integrator addresses this by embedding a secure, governed communication layer directly into the platform architecture.

By combining end-to-end encryption, backend-controlled access, and SDK-delivered communication runtime, it enables platforms to introduce real-time communication while maintaining control, reducing exposure, and aligning with modern security expectations.

2. The Platform Communication Risk Landscape

Modern digital platforms increasingly function as systems of engagement rather than static systems of record. Communication between users, clients, operators, and stakeholders is no longer peripheral — it is central to the operation of the platform.

However, communication is frequently implemented through mechanisms that are structurally misaligned with platform governance.

Many platforms rely on:

- External messaging applications
- Email-based workflows
- Internally developed real-time communication systems

Each approach introduces distinct limitations.

External messaging tools move communication outside the platform boundary. While convenient, they introduce a loss of visibility, control, and governance. Communication data is no longer managed within the platform's security model, and user interactions become fragmented across systems.

Email-based workflows lack real-time control and create persistent records that may increase exposure over time. They are also difficult to govern at scale, particularly in environments where access control and lifecycle management are critical.

Internally developed communication systems introduce substantial engineering and security challenges. Real-time communication infrastructure requires expertise in signalling, encryption, scalability, and device-level behaviour. Maintaining such systems becomes an ongoing operational burden.

In regulated or high-value environments, these limitations translate into:

- Increased compliance exposure
- Reduced control over sensitive interactions
- Higher operational risk
- Delays in delivering secure functionality

Pryvate Integrator addresses this landscape by embedding communication as a **controlled subsystem within the platform**, rather than as an external dependency or internally built liability.

3. What Pryvate Integrator SDK Is

Pryvate Integrator SDK is composed of two coordinated layers that operate together as a unified communication subsystem.

3.1 Backend Integration Layer (Control Plane)

The backend integration layer forms the control plane of the system. It is responsible for all operations that require trust, authority, and governance.

Through this layer, the integrator's backend system interacts with Pryvate via secure server-to-server API calls. These interactions include:

- Registering users into the communication environment
- Generating access credentials and session tokens
- Controlling communication lifecycle and participation

All critical operations are performed within the backend environment, ensuring that sensitive actions are not exposed to the client application.

This approach maintains a clear separation between trusted backend logic and untrusted client environments.

3.2 SDK Runtime Layer (Client Application)

The SDK runtime layer is embedded within the host application and provides the communication functionality experienced by users.

This includes:

- Messaging interfaces
- Voice and video communication
- Real-time state management
- Notification handling

The SDK operates entirely within the application's context. It does not replace the application's user experience, nor does it introduce a separate product layer.

Instead, it enables communication to exist as a natural extension of the application's existing functionality.

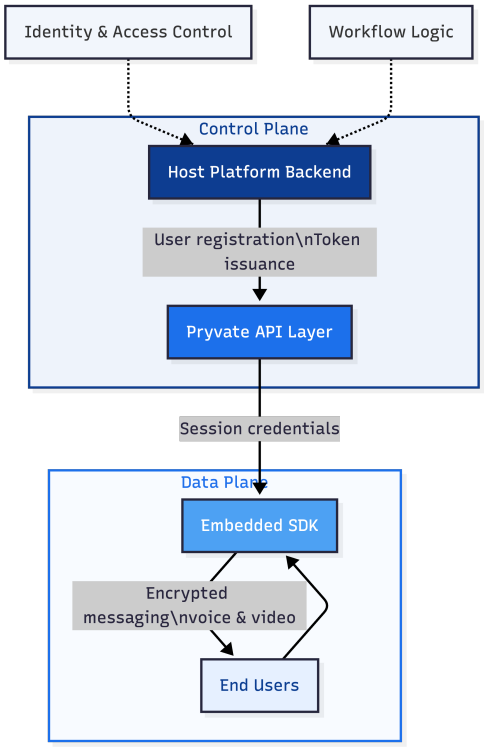
Key Architectural Principle

The integrator's platform remains:

- The system of record for user identity
- The authority on access and permissions
- The controller of workflow and session logic

Pryvate provides the communication capability, but does not assume ownership of the platform or its users.

High-Level Architecture Overview



4. Operational Workflow

Pryvate Integrator follows a structured lifecycle model that aligns communication with platform control.

4.1 Platform Integration

The integrator establishes a connection between their backend system and Pryvate APIs.

This process does not require the deployment of real-time communication infrastructure. Instead, the integrator leverages Pryvate's existing communication layer while maintaining control through backend integration.

4.2 Controlled User Registration

Users are registered into the communication environment through the integrator's backend.

Each user is assigned a unique identifier originating from the platform. Registration is performed via secure API calls, ensuring that onboarding is governed by backend logic rather than user-driven actions.

This replaces uncontrolled communication onboarding with a structured, policy-driven model.

4.3 Credential Issuance

Once registered, the backend generates communication credentials for each user.

These credentials include:

- Access tokens
- Session configuration parameters
- Device identifiers

They are used exclusively for SDK initialisation and are controlled by the backend.

4.4 SDK Initialisation

Within the application, the SDK is initialised during app startup.

The communication runtime is prepared, session credentials are applied, and communication services are activated.

This process establishes a secure communication environment within the application.

4.5 Active Communication

Users interact with communication features directly within the application.

Messaging, voice, and video capabilities are integrated into the platform's user interface and workflows. Users do not need to switch to external tools, and communication remains contained within the system.

4.6 Session Lifecycle Control

The integrator retains control over the lifecycle of communication.

This includes:

- When communication is enabled
- Who can participate
- How sessions are initiated and terminated

Communication becomes a controlled component of the platform, rather than an uncontrolled extension.

5. Security Architecture & Encryption Model

Pryvate Integrator is built on an encryption-first architecture designed to protect the confidentiality and integrity of communication.

This ensures that compromise of infrastructure components does not result in exposure of communication content.

5.1 End-to-End Encryption

All communication — including messages, voice calls, video sessions, and file transfers — is encrypted at the device level before transmission.

Decryption occurs only on the recipient's device. Communication infrastructure does not have access to plaintext content.

This ensures that communication remains protected even in the event of network or infrastructure compromise.

5.2 Cryptographic Framework

The system uses modern elliptic curve cryptography for secure key exchange.

Each communication session establishes independent cryptographic material, including:

- Ephemeral session keys
- Key rotation during extended sessions
- Forward secrecy

Forward secrecy ensures that the compromise of long-term keys does not expose historical communications.

5.3 Trust Model

Pryvate operates under a zero-trust assumption:

- Network infrastructure is not trusted
- Transport layers are not trusted
- Only endpoint devices are trusted with decrypted content

This model reduces reliance on infrastructure-level security and minimises structural exposure.

5.4 Key Management Model

Keys are generated and managed at the device level.

- Private keys remain on device
- No server-side key storage exists
- Session keys are ephemeral and not reused

This eliminates the possibility of centralised key compromise.

5.5 Metadata Exposure

While encryption protects content, metadata exposure remains a relevant consideration.

Pryvate is designed to minimise metadata retention and avoid behavioural profiling. Metadata is limited to operational requirements such as routing and service reliability.

Communication content is not used for analytics, profiling, or monetisation.

5.6 Threat Model

The architecture is designed to mitigate:

- Network interception
- Man-in-the-middle attacks
- Infrastructure-level access
- Communication data exposure

However, no system can eliminate risks associated with compromised authenticated endpoints or deliberate data exfiltration by authorised users.

6. Governance & Platform Control Alignment

Pryvate Integrator introduces governance at the communication layer without displacing control from the host platform.

Rather than acting as a separate system that users access independently, Pryvate operates within the boundaries defined by the integrator's architecture. This ensures that communication is subject to the same identity, access, and policy controls that govern the rest of the application.

6.1 Platform Authority

The integrator retains full authority over:

- User identity and authentication
- Access permissions and role assignment
- Session initiation and termination
- Communication availability within workflows

Pryvate does not introduce an alternative identity model or external user management system. All users originate from, and remain governed by, the host platform.

This ensures that communication is not an independent layer of interaction, but an extension of the platform's existing control model.

6.2 Backend-Controlled Access

All communication capability is enabled through backend-issued credentials.

Users do not gain access to communication features through client-side actions alone. Instead:

- The backend registers users into the communication layer
- Access tokens are generated under backend control
- Session credentials are issued per user

This model ensures that communication access is always aligned with platform policy and cannot be initiated outside defined system rules.

6.3 Communication as a Governed Capability

In many systems, communication is treated as a user-driven feature. In Pryvate Integrator, it is treated as a governed capability.

This distinction is important.

Communication can be:
- enabled or disabled based on workflow context
- restricted to specific user roles or groups
- aligned with operational processes (e.g. transaction flows, case handling, client interaction)

This allows communication to exist as part of the platform's logic rather than as an uncontrolled interaction layer.

6.4 Lifecycle Alignment

Communication access follows the lifecycle defined by the platform.

When a user is:
- onboarded → communication capability is provisioned
- active → communication operates within defined permissions
- restricted or revoked → communication access is removed

This ensures that communication does not persist beyond the user's authorised lifecycle within the platform.

6.5 Separation of Control and Content

Pryvate's architecture maintains a clear separation between:
- **Control Plane** (identity, access, session governance)
- **Data Plane** (encrypted communication content)

The integrator governs the control plane through backend systems, while the data plane remains encrypted and inaccessible to infrastructure.

This separation ensures that governance can be enforced without exposing communication content.

6.6 Alignment with Platform Policy and Regulation

Pryvate Integrator does not replace compliance frameworks or organisational policy. Instead, it enables those policies to be enforced at the communication layer.

By embedding communication within the platform:
- Access control is consistent across all user actions
- Communication does not bypass platform governance
- Data remains within defined system boundaries
- Security controls are applied uniformly

This supports alignment with modern principles such as:
- security by design
- controlled access

- data minimisation
- system accountability

6.7 Governance Outcome

The result of this model is that communication is no longer an external or loosely governed activity.

Instead, it becomes: **a controlled, policy-aligned component of the platform architecture**

This reduces structural risk while enabling real-time interaction as part of the platform's core functionality.

7. SDK Capabilities

The SDK provides a complete communication layer suitable for modern applications.

This includes:

- Real-time messaging (1:1 and group)
- Voice and video communication
- Media and file sharing
- Presence and typing indicators
- Push notification support

The SDK also supports both pre-built interfaces and fully custom implementations.

8. Integration Flexibility

Pryvate Integrator supports different levels of integration depending on platform requirements.

A rapid integration approach allows platforms to deploy communication quickly using pre-built UI components and default flows.

A custom integration approach enables full control over the user experience, allowing communication to be tightly integrated into existing workflows and interface design.

This flexibility ensures that communication can be introduced without compromising product architecture or user experience.

9. Risk Exposure Comparison

The purpose of this comparison is not to suggest that existing communication methods are inherently insecure, but to highlight the structural differences between unmanaged or externally dependent communication models and a controlled, embedded communication layer.

For many platforms, communication is introduced in ways that sit outside the core system architecture. This may include reliance on external messaging applications, email-based workflows, or fragmented communication tools. While these approaches can provide functionality, they often introduce governance gaps that are not immediately visible at the feature level.

These gaps typically emerge in areas such as user lifecycle control, data boundary management, and consistency of security enforcement.

Pryvate Integrator addresses this by embedding communication directly within the platform, allowing communication to be governed in alignment with the platform's existing identity, access, and policy frameworks.

The table below outlines the difference in operational risk posture between external or unmanaged communication approaches and an embedded Pryvate Integrator deployment.

Risk Category	Without Embedded Communication Layer	With Pryvate Integrator SDK
Communication Control	Communication occurs across external or fragmented tools, often outside platform visibility	Communication is embedded within the platform and governed by backend logic
User Governance	Access is inconsistent and often dependent on user behaviour or external systems	Access is defined and enforced by the platform backend
Data Boundary	Communication data may reside outside the platform boundary	Communication remains within the platform's controlled environment
Lifecycle Control	Onboarding and offboarding are informal or delayed	User access and communication capability are governed structurally
Security Model	Varies by tool and implementation	Consistent encryption-first architecture across all communication
Operational Visibility	Limited visibility into communication participation and flow	Communication participation is aligned with platform-defined identity and roles
Dependency Risk	Reliance on third-party tools introduces external dependencies	Communication capability is embedded and controlled internally
Regulatory Exposure	Increased exposure due to distributed communication channels	Reduced exposure through contained and governed communication layer

This comparison reflects a shift from communication as an external utility to communication as a controlled component of platform architecture.

10. Integration Model Context

Pryvate Integrator supports multiple integration approaches depending on platform architecture and deployment requirements.

The SDK model described in this document is designed for mobile applications where communication must be embedded directly within the user experience, with session control and identity governed by the host platform.

Alternative models may include backend-driven communication or web-based implementations, depending on system design.

11. Strategic Positioning

Pryvate Integrator is not a messaging application and does not operate as a collaboration platform.
It is a communications infrastructure layer designed for integration into third-party systems, enabling secure messaging, voice, and video capability to be delivered entirely within the host application.

Its purpose is to allow platforms to introduce real-time communication without creating additional operational risk, engineering complexity, or loss of control over users, data, or workflows.

12. Commercial & Deployment Considerations

Pryvate Integrator is delivered as an SDK-based integration.

It does not require:

- infrastructure build
- telecom stack development
- real-time communication system design

It can be deployed within existing applications and scaled alongside platform growth.

13. Technical Evaluation Resources

To support technical evaluation, the following materials are available:

- Sample SDK application
- React Native integration guide
- Backend integration guide

These resources allow engineering teams to assess integration flow, runtime behaviour, and implementation complexity.

14. Final Positioning

Pryvate Integrator transforms communication from an external dependency or engineering burden into a controlled, embedded infrastructure layer.

For platforms operating in environments where communication carries value, sensitivity, or regulatory exposure, the question is no longer whether communication is required — but whether it is governed.

Pryvate enables that governance without requiring platforms to build, operate, or maintain communication infrastructure themselves.

Annex A Technical Security Architecture

A.1 System Overview & Trust Model

Pryvate Integrator operates under a zero-trust communications model, designed to minimise reliance on infrastructure trust and reduce exposure across network and system layers.

The system assumes that:

- Network infrastructure is not trusted
- Transport layers are not trusted
- Communication routing systems are not trusted with plaintext content
- Only authenticated endpoint devices participating in a communication session are trusted with decrypted content

The architecture is divided into two logical planes:

Data Plane

The data plane is responsible for the transmission of encrypted communication payloads, including:

- message content
- voice streams
- video streams
- file transfers

All data within this plane is encrypted prior to transmission and remains encrypted throughout transit and routing.

Control Plane

The control plane is responsible for:

- user identity association (as provided by the integrator backend)
- session establishment and signalling
- message routing metadata
- presence and connection state
- SDK session configuration

The control plane does not have access to decrypted communication content.

This separation ensures that communication content remains protected even in the event of control plane compromise, and that governance can be applied without exposing sensitive data.

This model ensures that compromise of infrastructure components does not result in exposure of communication content.

A.2 Cryptographic Architecture

A.2.1 Key Exchange Mechanism

Pryvate uses elliptic curve cryptography based on the X25519 (Curve25519 family) algorithm for Elliptic Curve Diffie–Hellman (ECDH) key exchange.

This provides:

- strong security relative to key size
- efficient computation suitable for mobile environments

- widely accepted and well-studied cryptographic primitives

Each communication session establishes independent cryptographic material using this mechanism.

A.2.2 Session Key Establishment

For each communication session:

1. A key exchange handshake is performed between participating devices
2. Ephemeral session keys are derived
3. Encryption keys for message payloads and media streams are generated

Session keys are:

- unique per communication session
- not reused across unrelated sessions
- destroyed after session termination

This ensures that each interaction is cryptographically isolated.

A.2.3 Forward Secrecy

Pryvate implements forward secrecy principles.

This means:

- Compromise of long-term identity keys does not expose historical communication content
- Each session uses independently derived cryptographic material
- Historical communications remain protected even if future credentials are compromised

This significantly reduces the impact of key compromise scenarios.

A.2.4 Media Encryption

Voice and video communication streams are encrypted at the application layer prior to transmission.

This ensures that:

- transport infrastructure cannot access plaintext media
- encrypted streams remain unintelligible to intermediate network observers
- confidentiality is preserved independently of transport-layer security

A.2.5 File Encryption

Files transmitted within communication sessions are:

- encrypted before transmission
- bound to session-level encryption keys
- not stored in plaintext on intermediary systems

This ensures that file-based communication is subject to the same security model as messaging and media.

A.3 Key Management Model

Pryvate Integrator follows a device-centric key management model.

A.3.1 Identity Keys

- Identity keys are generated on-device
- Private keys remain on-device
- Private keys are never transmitted to or stored by infrastructure systems

This ensures that decryption capability is restricted to endpoint devices.

A.3.2 Session Keys

Session keys are:
- derived during session establishment
- ephemeral in nature
- not stored long-term

Session key rotation occurs:
- at session initiation
- during extended communication sessions
- upon reauthentication events

A.3.3 Key Storage Principles

- No key escrow is implemented
- No server-side private key storage exists
- No infrastructure-level decryption capability is present

This eliminates centralised points of cryptographic compromise.

A.4 Metadata Exposure Model

While encryption protects communication content, metadata exposure remains a relevant consideration.

Pryvate Integrator is designed to:
- minimise metadata storage
- avoid content indexing
- avoid behavioural profiling
- avoid monetisation of communication patterns

Metadata retained is limited to:
- routing information
- delivery state
- session coordination requirements

Metadata does not include:
- message content
- file content
- encryption keys

Communication data is not used for analytics, profiling, or advertising.

A.5 Threat Model

The architecture is designed to mitigate a defined set of structural threats.

A.5.1 Network-Level Threats

- Passive network interception
- Man-in-the-middle (MITM) attempts
- Transport-layer compromise

End-to-end encryption ensures that intercepted data remains unintelligible.

A.5.2 Infrastructure-Level Threats

- Server-side content access
- Insider infrastructure misuse

Because communication content is encrypted at the application layer and private keys are not held by infrastructure, plaintext access is not possible.

A.5.3 Platform Boundary Risks

- Communication leaving platform control
- External messaging dependencies

Pryvate mitigates these by embedding communication directly within the host application.

A.5.4 Endpoint Risk

Residual risks include:

- compromise of authenticated endpoint devices
- deliberate data exfiltration by authorised users
- screenshot or screen recording

These risks are inherent to all endpoint-based systems and are not fully mitigable by communication infrastructure alone.

A.5.5 Explicit Scope Limitations

No system can mitigate:

- physical compromise of an unlocked device
- malicious actions by authorised users
- coercion-based attacks

Explicit scope definition is essential for accurate risk assessment.

A.6 Identity & Access Control Model

Access to the communication layer is governed entirely by the integrator's backend systems.

Key characteristics include:

- user identity originates from the host platform
- communication access is granted through backend registration

- session credentials are issued per user
- access is enforced through token-based authentication

User lifecycle is controlled by the platform, not by Pryvate.

This ensures that communication capability aligns with platform-defined identity and access policies.

A.7 Infrastructure Trust Boundaries

The architecture enforces strict trust boundaries:

- Infrastructure can route encrypted data
- Infrastructure cannot decrypt communication content
- Infrastructure does not access private keys
- Infrastructure does not profile or monetise communication behaviour

This separation limits exposure in the event of infrastructure compromise.

A.8 Control vs Data Plane Separation

The system maintains a clear distinction between:

Control Plane

- identity association
- session establishment
- routing and signalling

Data Plane

- encrypted communication payloads

The control plane operates without access to decrypted content.

This separation ensures that governance and communication security can coexist without compromise.

A.9 Compliance Alignment (Technical Perspective)

From a technical architecture perspective, Pryvate Integrator supports:

- **Data minimisation** Communication content is encrypted and not accessible to infrastructure
- **Integrity & confidentiality** End-to-end encryption protects communication during transmission and storage
- **Access control** Backend-governed identity and token-based access
- **Security by design** Encryption-first architecture embedded at the communication layer
- **Accountability support** Communication participation is aligned with platform identity and access models

Responsibility for regulatory compliance, lawful basis, and policy enforcement remains with the integrator, as appropriate for their deployment context.