Admin Control Plane

Doc type: technical · **Version:** v0.1 · **Status:** published · **Module slug:** admin-control-plane
Exported: 2026-05-15 11:10 UTC · **By:** anonymous

Admin Control Plane – Technical Specification

1. Module Purpose & Scope (Authoritative)

The Admin Control Plane (ACP) is Primoro's internal, non-customer-facing operational platform for managing the full SaaS tenant lifecycle. It enables Primoro staff to onboard, provision, entitle, bill, support, and monitor customer tenants from a single, strictly audited internal environment. The ACP exists so that Primoro can operate as a multi-tenant SaaS business rather than a collection of individually managed installations.

The ACP is not part of Primoro CORE and does not run inside customer tenants. It operates as a dedicated internal admin tenant with elevated but strictly governed capabilities.

It governs:

- The prospect-to-customer lifecycle, including pipeline management, onboarding workflows, and tenant provisioning.
- Module and feature entitlements, billing orchestration, PMS integration management, and telephony configuration.
- Just-in-time (JIT) support access, tenant health monitoring, and full internal auditability across all admin actions.

It explicitly does not:

- Own or execute customer operational workflows — those are owned by their respective CORE modules.
- Store, process, or make decisions on clinical data — clinical data ownership belongs to clinical-domain modules within customer tenants.
- Provide analytics dashboards for customers or any patient-facing experience.
- Govern day-to-day practice configuration after go-live — post-go-live practice settings are managed within the customer tenant by the relevant CORE modules.

2. Ownership & Responsibilities

2.1 Admin Control Plane IS Responsible For

- Prospect → customer lifecycle management end-to-end (pipeline, onboarding, provisioning, activation).
- Module-level entitlement decisions and feature flag control, including regulatory-consequence extensions.
- PMS integration management, telephony provisioning, and billing orchestration for all tenants.
- Just-in-time support access with time-boxed sessions, approval workflow, and automatic expiry.
- Tenant health monitoring, alerting, and auto-generation of support tasks.
- Full auditability of every admin action — who, when, which tenant, what changed, old value → new value, and reason where required — using the canonical `SecurityEvent` model defined by Security & Privacy.

- Emitting all audit events to the Audit & Compliance module.

2.2 Admin Control Plane IS NOT Responsible For

- Customer-tenant operational workflows — owned by individual CORE modules within each customer tenant.
- Clinical decision support or any clinical data — owned by clinical-domain CORE modules.
- Customer-facing analytics or reporting — owned by the Analytics module.
- Patient-facing experiences — owned by patient-facing CORE modules.
- Identity management and RBAC within customer tenants — owned by Access Manager.
- Digital form content and rendering — owned by Digital Forms; ACP only controls whether the Digital Forms integration is enabled per tenant.

3. Core Objects (Normative)

3.1 Tenant (Canonical Artefact)

A Tenant is a governed digital artefact representing a provisioned or in-progress customer account within the Primoro platform.

Minimum required fields:

- `TenantId` — UUID, immutable primary key
- `Status` — current lifecycle state (see §3.2)
- `Region` — hosting region
- `ModulesEnabled` — list of entitled module keys
- `Clinics[]` — child clinic records within the tenant
- `Groups[]` — multi-clinic group memberships
- `BillingAccountId` — FK to Finance Centre billing account
- `TelephonyConfig` — telephony provisioning configuration
- `CreatedAt` — UTC timestamp, immutable
- `AuditTrail` — immutable append-only log of all state and configuration changes

3.2 Tenant State Machine (Authoritative)

States:

- **Prospect** — lead entered into the Admin Pipeline Manager; no tenant provisioned.
- **Onboarding** — proposal accepted; Onboarding Workflow Engine activated; tenant not yet live.
- **Provisioning** — Tenant Builder running; infrastructure and identity realm being created.
- **Live** — tenant fully provisioned, smoke tests passed, go-live handoff complete.
- **Supported** — active support session in progress against the tenant.
- **Suspended** — tenant access restricted, typically due to billing or compliance issue.
- **Billed** — normal billing-active state; overlaps with Live in most cases.
- **Monitored** — Tenant Health Monitor actively tracking the tenant.

- **Decommissioned** — tenant offboarded; data retained per retention policy.

Rules:

- All state transitions are auditable and timestamped.
- A tenant cannot transition from Provisioning to Live until all smoke tests pass.
- A tenant cannot be deleted; it must be moved to Decommissioned.
- Only a Provisioning Engineer or SuperAdmin may initiate the Provisioning → Live transition.
- Support Mode access (Supported state overlay) does not alter the tenant's primary lifecycle state.

3.3 OnboardingTask (Canonical Artefact)

An OnboardingTask is a governed unit of work within a structured onboarding workflow, assigned to a role with a defined SLA.

Minimum required fields:

- `TaskId` — UUID
- `TenantId` — FK to parent Tenant
- `Stage` — onboarding stage this task belongs to
- `SLA` — target completion deadline
- `AssignedRole` — admin role responsible for completion
- `Status` — current task state
- `CreatedAt` — UTC timestamp
- `AuditTrail` — immutable

3.4 Entitlement (Canonical Artefact)

An Entitlement is a governed record controlling whether a given module or feature is active for a tenant.

Minimum required fields:

- `ModuleKey` — canonical identifier for the module or feature
- `Enabled` — boolean
- `Scope` — `Tenant` or `Clinic`
- `EffectiveDate` — UTC date the entitlement came into force
- `EnabledBy` — actor who set the entitlement
- `AuditTrail` — immutable; mandatory reason field for regulated extensions (see §5.5)

3.5 SupportSession (Canonical Artefact)

A SupportSession is a governed record representing a time-boxed elevated access grant to a customer tenant.

Minimum required fields:

- `SessionId` — UUID
- `TenantId` — FK to target Tenant
- `ApprovedBy` — admin identity of the approving actor
- `ExpiresAt` — UTC timestamp; hard expiry enforced by system

- `AuditLog` — immutable activity log for the duration of the session

4. Administrative Functional Areas

4.1 Admin Pipeline Manager

The Admin Pipeline Manager is an internal sales pipeline forked from — and not sharing data models with — any customer-facing pipeline module.

The module MUST:

- Support qualification, scoring, and handover of prospects.
- Enforce task assignments, SLAs, and note capture per pipeline stage.
- Trigger onboarding upon conversion (proposal acceptance or manual admin initiation).

The module MAY:

- Accept prospect intake from web forms, referrals, chat, phone, and assisted manual entry.
- Support group-expansion events as an onboarding trigger.

The module MUST NOT:

- Share pipeline logic or data with customer-tenant pipeline modules.
- Expose pipeline data to customer tenants.

4.2 Onboarding Workflow Engine (Authoritative)

The Onboarding Workflow Engine manages structured, stage-gated onboarding from proposal acceptance to go-live handoff.

The module MUST:

- Provide task templates, SLA enforcement, and validation gates at each onboarding stage.
- Collect required data via Digital Forms integration.
- Block progression past a validation gate until all required inputs are satisfied.
- Deliver a structured go-live handoff record on completion.

The module MAY:

- Allow admins to customise task templates per onboarding workflow.

4.3 Setup Wizard (Authoritative)

The Setup Wizard provides post-provisioning guided configuration for a newly created tenant, sequenced Core-first with conditional steps driven by the tenant's active entitlements.

The module MUST:

- Present all configuration in a conditional, entitlement-aware step sequence.
- Auto-save progress and validate each step before allowing progression.
- Record all configuration changes in the audit log against the tenant.

Care Plan Subscriptions conditional step: When the Care Plan Subscriptions module is entitled for a tenant, the Setup Wizard MUST present a dedicated conditional step covering:

- Enabling or disabling Care Plan Subscriptions at practice level.
- Toggling the Digital Forms integration for Care Plan logic (i.e. whether digital forms are aware of and interact with Care Plan state).
- Selecting which form types trigger Care Plan interactions (e.g. consent forms, treatment plan acceptance forms).

This step MUST be skipped entirely when the Care Plan Subscriptions entitlement is absent. All selections made within this step MUST be recorded in the audit log as configuration changes against the tenant.

Knowledge, Training & Learning conditional step: When the Knowledge, Training & Learning (KTL) module is entitled for a tenant, the Setup Wizard MUST present a dedicated conditional step covering:

- Enabling or disabling the LMS capability at practice level.
- Configuring external CPD integration endpoints and credentials, where applicable.
- Selecting which Primoro-published content libraries are available to the tenant's practitioners.
- Enabling or disabling learning pathway assignment for staff roles within the tenant.

This step MUST be skipped entirely when the KTL entitlement is absent. All selections made within this step MUST be recorded in the audit log as configuration changes against the tenant. KTL entitlement is managed through the Entitlement Manager following the standard module entitlement pattern (see §4.5); it does not carry regulatory-consequence status unless a specific learning pathway is designated as mandatory for a regulated extension (e.g. Medication & Controlled Drugs induction training).

4.4 Tenant Builder — Provisioning Engine (Authoritative)

The Tenant Builder creates and activates a new customer tenant deterministically and idempotently.

The module MUST:

- Create the tenant record, identity realm, core module seed, PMS connector initialisation, and communication channel configuration as discrete, recoverable steps.
- Execute smoke tests before marking a tenant as Live.
- Make all failed provisioning steps visible and individually recoverable without requiring a full re-run.

The module MUST NOT:

- Mark a tenant Live if any smoke test has failed.
- Leave orphaned tenant artefacts on partial failure.

All provisioning steps are idempotent: re-running a step that has already succeeded MUST NOT produce duplicate artefacts.

4.5 Entitlement Manager (Authoritative)

The Entitlement Manager controls which modules and features are active for each tenant, with per-clinic scoping where required.

The module MUST:

- Support module-level entitlements, feature flags, per-clinic enablement, and rollout controls.
- Keep entitlement state synchronised with the Finance Centre billing record.

Regulated extension — Medication & Controlled Drugs (Inventory & Compliance Manager):

The Medication & Controlled Drugs extension is an optional capability within the Inventory & Compliance Manager module. It MUST be enabled as a separate, per-tenant entitlement toggle and is never active by default.

Because enabling this extension activates the Controlled Drugs Register, enforces immutable audit entries for all CD transactions, and imposes controlled-storage configuration requirements, the following rules apply:

- Only a Provisioning Engineer or SuperAdmin may enable or disable this entitlement.
- Enablement MUST be recorded as an explicit audit event with the approving admin's identity and a mandatory reason field; the audit entry MUST be rejected at write time if the reason field is absent or empty.
- The entitlement is scoped at tenant level but MAY be further restricted to specific clinics within a group.
- Disabling the extension after it has been active MUST be gated behind a confirmation step that warns of regulatory implications; the action is logged regardless of whether the operator proceeds.
- Billing sync with the Finance Centre MUST reflect whether this extension is active, as it may attract separate contractual or licensing obligations.

This pattern — explicit gating, elevated role requirement, mandatory reason, and billing reflection — is the reference model for any future extension that carries regulatory consequence.

4.6 PMS Connector Registry

The PMS Connector Registry is the central register for all PMS integrations across all tenants.

The module MUST:

- Maintain token scoping, labelling, and sandbox-vs-production separation per connector.
- Run health checks and expose diagnostics per connector.
- Monitor webhook delivery and usage.

Traffic class and gateway governance: All PMS connector API calls to external provider systems MUST be routed through the External Provider API Gateway & Rate Governance layer. Each connector registered in the PMS Connector Registry MUST declare a traffic class at registration time; unclassified connectors MUST be rejected by the registry. The traffic class declaration is stored alongside the connector record and is included in all health check and diagnostic outputs. This requirement applies equally to sandbox and production connector configurations. See §6 Integration Contracts for the outbound gateway contract.

4.7 Finance Centre

The Finance Centre orchestrates billing and payment for all tenants.

The module MUST:

- Manage subscription billing via GoCardless (mandates, retries, dunning).
- Manage transactional payments via DNA Payments.
- Support usage-based charging and accounting exports.
- Reflect entitlement changes (including regulated extensions) in billing records.
- Restrict revenue visibility to admin roles only.

Lab Manager accounting exports: Where a tenant has the Lab Manager module entitled, the Finance Centre MUST support accounting export triggers scoped to lab transaction data. The export model and trigger configuration for lab-specific accounting are declared in the Lab Manager module spec; the Finance Centre's

responsibility is to honour the export trigger events emitted by the ACP on behalf of the entitled module.

4.8 Telephony Centre

The Telephony Centre manages all telephony provisioning and configuration for customer tenants.

The module MUST:

- Provision numbers, configure routing rules, and manage after-hours logic per tenant.
- Enable or disable the AI Phone Assistant per tenant.
- Capture call transcription and sentiment data.
- Enforce PCI voice masking.
- Expose telephony analytics to admin roles.

4.9 Multi-Clinic Group Manager

The Multi-Clinic Group Manager governs tenant hierarchies where a customer operates more than one clinic.

The module MUST:

- Support group creation, clinic hierarchy, and configuration inheritance.
- Manage group-level billing and group-level permissions.
- Provide cross-site visibility for admin roles.

4.10 Support Mode — JIT Access (Authoritative)

Support Mode provides just-in-time, time-boxed elevated access to a customer tenant for Primoro support staff.

The module MUST:

- Require an approval workflow before access is granted.
- Enforce a hard session expiry at `ExpiresAt`; no session may be extended without re-approval.
- Capture a full activity audit log for every action taken within the session.
- Prevent any standing privileged access; no permanent elevated grants are permitted.

The module MUST NOT:

- Allow a Support Engineer to open a session without an approved SupportSession record.
- Retain elevated access past the `ExpiresAt` timestamp under any circumstances.

SecurityEvent emission (normative): Support Mode MUST emit the following canonical SecurityEvents to the Audit & Compliance event stream at the points indicated. These event types are defined in Security & Privacy §3.2 and MUST be emitted with all mandatory fields populated; emission of an event with missing mandatory fields MUST cause the session operation to be rejected and rolled back.

- **`SupportSessionGranted`** — emitted immediately upon approval and session creation, before access is made available to the Support Engineer. Mandatory fields: `SessionId`, `TenantId`, `ApprovedBy`, `ExpiresAt`. The `Actor` field MUST identify the approving admin; the `Target` field MUST reference the tenant in the form `Tenant:<TenantId>`.
- **`SupportSessionExpired`** — emitted when the session reaches `ExpiresAt` via system-enforced hard expiry, or when a Support Engineer or SuperAdmin closes the session early. Mandatory fields: `SessionId`, `TenantId`, `ApprovedBy`, duration (calculated as `ExpiresAt` minus session start, or actual close time if closed

early), and a summary of actions taken within the session. Where the session is closed early by an admin, the closing actor's identity MUST also be recorded.

Both events are ACP-specific extensions of the canonical `SecurityEvent` enumeration and MUST be declared in the Security & Privacy module's extension registry (see §8 and Open Question 8).

4.11 Tenant Health Monitor

The Tenant Health Monitor provides continuous observability of tenant operational health and escalates issues automatically.

The module MUST:

- Monitor PMS sync failures, payment and billing issues, telephony degradation, form and workflow errors, and appointment sync issues.
- Auto-generate support tasks in the Task Engine when monitored thresholds are breached.
- Surface active alerts to admin roles with sufficient context to act without requiring direct tenant access.

Task-creation contract: When a monitored threshold is breached, the Tenant Health Monitor MUST emit a task-creation event to the Task Engine conforming to the Task Manager canonical task schema. Each emitted task MUST include:

- `SourceModule` — set to `AdminControlPlane.TenantHealthMonitor`.
- `TenantId` — the tenant against which the health event was detected.
- `HealthEventType` — a structured classification of the triggering condition (e.g. `PmsSyncFailure`, `PaymentFailure`, `TelephonyDegradation`, `FormWorkflowError`, `AppointmentSyncFailure`).
- `Severity` — mapped from the health monitor's internal threshold breach level to the Task Manager's canonical severity scale.
- `AlertContext` — structured metadata providing sufficient detail for a Support Engineer to act without requiring direct tenant access.
- `AssignedRole` — the admin role responsible for resolution; defaults to `Support Engineer` unless the health event type mandates escalation to `SuperAdmin` or `Provisioning Engineer`.

Tasks generated by the Tenant Health Monitor are owned by the Task Engine from the moment of creation. Escalation of unresolved tasks (e.g. SLA breach re-assignment) is governed by Task Manager escalation rules, not by the ACP. The ACP MUST NOT duplicate task-creation events for the same health event; idempotency is enforced by including a deterministic `HealthEventId` derived from the tenant, event type, and detection timestamp.

AI Quality Monitor health signals: Where a tenant has the AI Quality Monitor module active, the Tenant Health Monitor MAY consume quality findings and zone-level indicators emitted by the AI Quality Monitor as additional health signals. The following integration rules apply:

- The ACP MUST NOT pull data directly from the AI Quality Monitor's internal store; signals MUST be received via the published event or webhook contract declared by the AI Quality Monitor.
- Only quality findings classified at or above a configurable severity threshold (configurable by SuperAdmin per tenant) MUST trigger a health alert or auto-generated task; lower-severity findings MUST be surfaced as informational context only.
- The specific quality metric types that the Tenant Health Monitor recognises from the AI Quality Monitor are: call quality degradation indicators, zone-level error rates, and AI-assisted workflow failure signals. Primoro platform engineering MUST maintain a mapping table between AI Quality Monitor event types and Tenant

Health Monitor `HealthEventType` values.

- This integration is optional at the tenant level; tenants without the AI Quality Monitor entitlement MUST NOT experience any degradation in Tenant Health Monitor behaviour.

5. Delivery Surfaces & Access (Authoritative)

5.1 Web Portal

The ACP is delivered exclusively through a dedicated internal web portal, accessible only to Primoro staff with a valid admin identity. The portal is not accessible from within any customer tenant and shares no navigation or session context with customer-facing surfaces.

5.2 Tablet App

Not applicable. The ACP has no tablet delivery surface.

5.3 Patient Mobile App

Not applicable. The ACP has no patient-facing delivery surface.

5.4 Engagement Signals

The Tenant Health Monitor emits operational alerts and auto-generated tasks as engagement signals for admin staff. Revenue and billing summaries are surfaced within the Finance Centre for Finance Admin roles only. Telephony analytics are surfaced within the Telephony Centre for Telephony Admin and SuperAdmin roles.

6. Integration Contracts

6.1 Inbound (this module consumes from)

From module	What	Contract
Access Manager	Admin identity and role assignments for ACP staff	Sync
Digital Forms	Form data collected during onboarding and Setup Wizard steps	Async
PMS Connector Layer	PMS health signals and connector status	Webhook / async
AI Phone Assistant	Telephony error and degradation signals	Webhook / async
AI Quality Monitor	Quality findings and zone-level health signals (optional; tenant-scoped)	Webhook / async

External Provider API Gateway & Rate Governance	Traffic class validation responses for PMS connector registration	Sync

6.2 Outbound (this module emits to)

To module	What	Contract
Access Manager	Tenant identity realm creation; admin role grants for JIT sessions	Sync
Task Engine	Auto-generated support tasks from Tenant Health Monitor and onboarding workflows, conforming to Task Manager canonical schema	Event
Digital Forms	Entitlement signal controlling whether Digital Forms integration is active per tenant	Event
Communication Hub	Notification triggers for state transitions (e.g. go-live, billing failure)	Event
PMS Connector Layer	Connector initialisation and configuration per tenant	Sync
Finance Centre	Entitlement changes requiring billing sync; lab accounting export triggers	Event
AI Phone Assistant	Enablement and configuration per tenant	Sync
Audit & Compliance	All admin audit events conforming to the SecurityEvent model, including SupportSessionGranted and SupportSessionExpired	Immutable event stream
Security & Privacy	ACP audit records conforming to the canonical SecurityEvent model	Event
External Provider API Gateway & Rate Governance	Traffic class declarations for all PMS connector registrations	Sync

6.3 PMS Boundary

The PMS Connector Layer owns all data exchange with third-party practice management systems. The ACP is responsible for initialising and registering PMS connectors during tenant provisioning, managing token scoping and sandbox-vs-production state, and surfacing connector health signals from the PMS Connector Layer in the Tenant Health Monitor. The ACP does not parse, transform, or store PMS appointment or clinical data.

All external provider API calls initiated via registered PMS connectors MUST be routed through the External Provider API Gateway & Rate Governance layer. The ACP MUST NOT register a connector without a validated traffic class declaration. Where the gateway rejects a traffic class declaration (e.g. unrecognised class, rate budget exceeded), the connector registration MUST fail with a descriptive error surfaced to the initiating admin, and the failure MUST be recorded in the audit log.

7. AI Boundaries (Non-Negotiable)

Module does not embed AI surfaces directly in its admin workflows, with the following bounded exception:

The Telephony Centre provisions and configures the AI Phone Assistant on behalf of customer tenants. In this context:

AI MAY:

- Capture call transcription and sentiment data for admin review.
- Surface analytics summaries of telephony activity to admin roles.

AI MAY NOT:

- Auto-configure telephony routing or after-hours logic without explicit admin approval.
- Bypass entitlement checks or audit logging.
- Make commitments on behalf of the practice or Primoro.
- Take action on a tenant without a human admin as the approving actor.

Where any AI-assisted action is recorded in the audit log, the `Actor` field MUST be set to `AI` per the SecurityEvent model defined by Security & Privacy, and the human approver MUST be separately identified.

8. Audit & Compliance

The ACP MUST conform to the canonical `SecurityEvent` model defined by Security & Privacy for all audit records. ACP-specific fields (`TenantId`, `OldValue`, `NewValue`, `Reason`) are carried as structured metadata alongside canonical SecurityEvent fields, not as replacements.

The system MUST log all of the following events:

Event	Mandatory fields beyond canonical SecurityEvent
Admin login / logout	—
Tenant state transition	`TenantId`, `OldValue` (previous state), `NewValue` (new state)
Entitlement enabled / disabled	`TenantId`, `ModuleKey`, `OldValue`, `NewValue`, `Reason`

Medication & Controlled Drugs extension enabled / disabled	`TenantId`, `ModuleKey`, `Reason` (mandatory; entry rejected if absent)
Care Plan Subscriptions Setup Wizard step completed	`TenantId`, changed configuration values
Knowledge, Training & Learning Setup Wizard step completed	`TenantId`, changed configuration values
Support Mode session opened (`SupportSessionGranted`)	`TenantId`, `SessionId`, `ApprovedBy`, `ExpiresAt`, `Reason`
Support Mode session expired or closed (`SupportSessionExpired`)	`TenantId`, `SessionId`, duration, summary of actions taken
Any action taken within a Support Mode session	`TenantId`, `SessionId`, full action detail
PMS connector initialised or reconfigured	`TenantId`, connector identity, `OldValue`, `NewValue`
PMS connector registration rejected (traffic class invalid)	`TenantId`, connector identity, rejection reason
Billing account created or entitlement billing sync triggered	`TenantId`, relevant billing identifiers
Lab accounting export triggered	`TenantId`, export scope, triggering entitlement
Provisioning step completed or failed	`TenantId`, step identity, outcome
AI Phone Assistant enabled / disabled	`TenantId`, `OldValue`, `NewValue`
Setup Wizard configuration change	`TenantId`, step, `OldValue`, `NewValue`
AI Quality Monitor health signal received and acted upon	`TenantId`, `HealthEventType`, severity, task created (boolean)

All audit log entries MUST be:

- Immutable and append-only.
- Exportable for compliance inspection.
- Rejected at write time if mandatory fields (including `Reason` where required) are absent.

Audit logs MUST be immutable and exportable for inspection.

The ACP SecurityEvent enumeration extends the canonical `EventType` values with the following ACP-specific types: `Provision`, `EntitlementChange`, `SupportSessionGranted`, `SupportSessionExpired`. These extensions MUST be documented in the Security & Privacy module's extension registry. Note: `SupportSessionGranted` and `SupportSessionExpired` supersede the previously referenced `SupportSessionOpen` and `SupportSessionClose` labels, aligning with the canonical names defined in Security & Privacy §3.2.

9. Access Control

Access to the ACP is governed by the Access Manager. The following roles are defined within the ACP's internal admin tenant:

Role	Capabilities
SuperAdmin	All capabilities including regulated extension enablement, JIT session approval, and billing visibility
Provisioning Engineer	Tenant creation, provisioning, entitlement management (including regulated extensions), go-live sign-off
Finance Admin	Billing account management, payment configuration, revenue visibility, accounting exports
Telephony Admin	Number provisioning, routing rules, AI Phone Assistant configuration, telephony analytics
Support Engineer	Support Mode session request (requires approval); read access to tenant health data
CSM	Onboarding workflow management, tenant health monitoring, read access to pipeline and entitlements
Sales	Admin Pipeline Manager read/write; no access to provisioned tenant data

Additional access rules:

- MFA is enforced for all admin identities without exception.
- Admin sessions are short-lived; no permanent elevated grants are permitted.
- Service-to-service automation uses scoped service tokens, not user identities.
- Only Provisioning Engineer or SuperAdmin may enable or disable regulated extensions (see §4.5).
- Support Mode sessions require explicit approval from a SuperAdmin or Provisioning Engineer before access is granted.
- No admin role has standing access to customer tenant data; all access is mediated through JIT Support Mode or provisioning workflows.

10. Integration Summary

- **Access Manager** — inbound admin identity and RBAC; outbound identity realm creation and JIT session grants.
- **Task Engine** — outbound auto-generated support and onboarding tasks conforming to the Task Manager canonical schema.
- **Digital Forms** — inbound form data collection during onboarding; outbound integration entitlement signals.

- **Communication Hub** — outbound notification events for tenant lifecycle state changes.
- **PMS Connector Layer** — inbound connector health signals; outbound connector initialisation and configuration.
- **Finance Centre** — outbound entitlement-driven billing sync; GoCardless and DNA Payments orchestration; lab accounting export triggers.
- **AI Phone Assistant** — outbound enablement and configuration; inbound telephony error signals.
- **Audit & Compliance** — outbound immutable admin audit event stream, including SupportSessionGranted and SupportSessionExpired events.
- **Security & Privacy** — ACP audit records conform to and extend the canonical SecurityEvent model; SupportSessionGranted and SupportSessionExpired types registered in the Security & Privacy extension registry.
- **Inventory & Compliance Manager** — Medication & Controlled Drugs extension entitlement is gated and managed by ACP.
- **Care Plan Subscriptions** — Setup Wizard conditional step is rendered only when this entitlement is active.
- **Knowledge, Training & Learning** — Setup Wizard conditional step is rendered only when this entitlement is active; LMS and CPD configuration surfaces managed via Entitlement Manager.
- **AI Quality Monitor** — inbound quality findings and zone-level health signals consumed by Tenant Health Monitor (optional; tenant-scoped).
- **External Provider API Gateway & Rate Governance** — all PMS connector registrations MUST declare a traffic class and route external provider calls through the gateway.

11. Explicit Non-Goals

- Running inside customer tenants — the ACP is always a separate, isolated internal tenant.
- Exposing any admin feature, screen, or data to customer users or patients.
- Skipping audit logging for any reason, including performance or urgency.
- Allowing permanent or standing elevated access to customer tenant data.
- Depending on Primoro CORE being enabled for ACP to operate.
- Providing customer-facing analytics — owned by the Analytics module.
- Clinical data storage or decision-making — owned by clinical-domain CORE modules.

12. Versioning & Governance

This specification is owned by: Primoro Platform Engineering (Cross-Cutting tier).

Changes to this spec require:

- Review by the Cross-Cutting module owner.
- Impact analysis across all declared related modules (see /propose), with particular attention to Security & Privacy (SecurityEvent model conformance), Access Manager (role changes), and Finance Centre (billing sync).
- Version bump: patch for clarifications, minor for new capabilities within existing scope, major for scope boundary changes.

All future changes must preserve:

- Tenant isolation — no customer tenant may access ACP modules or data.
- Least-privilege admin access — no standing elevated grants.
- Full auditability — no ACP action is exempt from audit logging.
- Lifecycle-driven design — every significant state is explicit, auditable, and recoverable where appropriate.

13. Build Contract (Engineering & QA)

13.1 Canonical Data Model

```
Tenant (
 TenantId            UUID         PRIMARY KEY,
 Status              VARCHAR      NOT NULL,  -- lifecycle state; see §3.2
 Region              VARCHAR      NOT NULL,
 ModulesEnabled      JSONB        NOT NULL DEFAULT '[]',
 Clinics             JSONB        NOT NULL DEFAULT '[]',
 Groups              JSONB        NOT NULL DEFAULT '[]',
 BillingAccountId    UUID         REFERENCES BillingAccount(BillingAccountId),
 TelephonyConfig     JSONB,
 CreatedAt           TIMESTAMPTZ  NOT NULL DEFAULT now()
)

OnboardingTask (
 TaskId              UUID         PRIMARY KEY,
 TenantId            UUID         NOT NULL REFERENCES Tenant(TenantId),
 Stage               VARCHAR      NOT NULL,
 SLA                 TIMESTAMPTZ  NOT NULL,
 AssignedRole        VARCHAR      NOT NULL,
 Status              VARCHAR      NOT NULL,
 CreatedAt           TIMESTAMPTZ  NOT NULL DEFAULT now()
)

Entitlement (
 EntitlementId       UUID         PRIMARY KEY,
 TenantId            UUID         NOT NULL REFERENCES Tenant(TenantId),
 ModuleKey           VARCHAR      NOT NULL,
 Enabled             BOOLEAN      NOT NULL DEFAULT FALSE,
 Scope               VARCHAR      NOT NULL CHECK (Scope IN ('Tenant', 'Clinic')),
 ClinicId            UUID,        -- populated when Scope = 'Clinic'
 EffectiveDate       DATE         NOT NULL,
 EnabledBy           UUID         NOT NULL  -- FK to admin user identity
)

SupportSession (
 SessionId           UUID         PRIMARY KEY,
 TenantId            UUID         NOT NULL REFERENCES Tenant(TenantId),
 ApprovedBy          UUID         NOT NULL,  -- FK to approving admin identity
 ExpiresAt           TIMESTAMPTZ  NOT NULL,
 CreatedAt           TIMESTAMPTZ  NOT NULL DEFAULT now()
)

AdminAuditEvent (
 EventId             UUID         PRIMARY KEY,
 EventType           VARCHAR      NOT NULL,  -- extends SecurityEvent.EventType
 Actor               VARCHAR      NOT NULL,  -- 'User' | 'System' | 'AI'
```

```
 ActorId             UUID,
 Target              VARCHAR      NOT NULL,  -- typed reference e.g. 'Tenant:<TenantId>'
 DeviceId            VARCHAR,
 Timestamp           TIMESTAMPTZ  NOT NULL DEFAULT now(),
 TenantId            UUID,
 OldValue            JSONB,
 NewValue            JSONB,
 Reason              TEXT         -- mandatory for regulated events; see §8
)

PmsConnectorRegistration (
 ConnectorId         UUID         PRIMARY KEY,
 TenantId            UUID         NOT NULL REFERENCES Tenant(TenantId),
 ConnectorType       VARCHAR      NOT NULL,
 TrafficClass        VARCHAR      NOT NULL,  -- mandatory; validated against gateway on registration
 Environment         VARCHAR      NOT NULL CHECK (Environment IN ('Sandbox', 'Production')),
 TokenScope          TEXT         NOT NULL,
 CreatedAt           TIMESTAMPTZ  NOT NULL DEFAULT now()
)
```

13.2 Core Behaviour Rules

1. Every ACP action MUST be scoped to an identified tenant or to the ACP internal tenant; no unscoped actions are permitted.
2. No admin identity may hold permanent elevated access to any customer tenant; all elevated access is time-boxed via Support Mode.
3. All provisioning steps MUST be idempotent; re-executing a completed step MUST NOT produce duplicate artefacts or errors.
4. A tenant MUST NOT be marked Live unless all smoke tests have passed and are recorded as passed in the audit log.
5. Any failed provisioning step MUST be surfaced in the admin UI as individually visible and individually retriable.
6. Audit logging is non-optional; if the audit write fails, the originating action MUST be rolled back.
7. An audit entry requiring a `Reason` field MUST be rejected at write time if the `Reason` is absent or empty.
8. A SupportSession MUST be hard-expired at `ExpiresAt`; no code path may extend a session without a new approval record.
9. Entitlement state and billing state MUST be kept in sync; an entitlement change MUST trigger a Finance Centre billing sync event within the same transaction or with a guaranteed delivery retry.
10. No customer tenant may access ACP modules, data, or API surfaces; cross-tenant data leakage MUST be tested as a negative acceptance criterion.
11. The Medication & Controlled Drugs entitlement MUST only be toggled by a Provisioning Engineer or SuperAdmin; any attempt by a lower role MUST be rejected with an appropriate error and logged as a failed access attempt.
12. The Setup Wizard Care Plan Subscriptions step MUST render if and only if the Care Plan Subscriptions entitlement is active for the tenant being configured.
13. The Setup Wizard Knowledge, Training & Learning step MUST render if and only if the KTL entitlement is active for the tenant being configured.
14. A `SupportSessionGranted` SecurityEvent MUST be emitted and successfully written before access is made available to the Support Engineer; if the event write fails, the session MUST NOT be opened.

15. A `SupportSessionExpired` SecurityEvent MUST be emitted on both system-enforced hard expiry and admin-initiated early closure; failure to write this event MUST be surfaced as an operational alert to SuperAdmin.

16. Every PMS connector registration MUST include a validated `TrafficClass`; registrations without a valid traffic class declaration MUST be rejected and the rejection logged.

17. Task-creation events emitted by the Tenant Health Monitor to the Task Engine MUST include a deterministic `HealthEventId` to prevent duplicate task creation for the same health event.

13.3 Configuration Surfaces

Surface	Configurable by	Location
Onboarding workflow templates and SLAs	Provisioning Engineer, SuperAdmin	Admin Control Plane → Onboarding Workflow Engine
Module and feature entitlements	Provisioning Engineer, SuperAdmin	Admin Control Plane → Entitlement Manager
Regulated extension toggles (Medication & Controlled Drugs)	Provisioning Engineer, SuperAdmin only	Admin Control Plane → Entitlement Manager (gated UI)
Knowledge, Training & Learning entitlement and LMS configuration	Provisioning Engineer, SuperAdmin	Admin Control Plane → Entitlement Manager; Setup Wizard (KTL step)
Billing rules and payment gateway configuration	Finance Admin, SuperAdmin	Admin Control Plane → Finance Centre
Lab accounting export trigger configuration	Finance Admin, SuperAdmin	Admin Control Plane → Finance Centre
PMS connector configuration (including traffic class declaration)	Provisioning Engineer, SuperAdmin	Admin Control Plane → PMS Connector Registry
Telephony routing and AI Phone Assistant	Telephony Admin, SuperAdmin	Admin Control Plane → Telephony Centre
Tenant health monitoring thresholds	SuperAdmin	Admin Control Plane → Tenant Health Monitor
AI Quality Monitor health signal severity threshold (per tenant)	SuperAdmin	Admin Control Plane → Tenant Health Monitor
Admin role assignments	SuperAdmin	Access Manager (via ACP surface)

13.4 Filtering & Views

The following standard filters and views MUST be supported in the ACP web portal:

- **Tenant list:** Filter by lifecycle status, region, entitlements active, health alert state, billing status, assigned CSM.
- **Onboarding task list:** Filter by stage, assigned role, SLA breach status, tenant.
- **Entitlement view:** Filter by module key, scope (Tenant / Clinic), enabled / disabled, tenant.
- **Support session log:** Filter by tenant, actor, date range, session status (active / expired).
- **Audit event log:** Filter by event type, actor, tenant, date range; exportable to CSV or structured format.
- **Tenant health dashboard:** Filter by alert type (PMS, payment, telephony, forms, appointments, AI quality), severity, tenant.

13.5 Module Extension Map

The ACP is designed to accommodate future regulated extensions following the Medication & Controlled Drugs reference pattern (§4.5). A new regulated extension is added by:

1. Defining a new `ModuleKey` in the Entitlement Manager.
2. Declaring which roles may toggle it (defaulting to Provisioning Engineer and SuperAdmin).
3. Specifying whether a mandatory `Reason` field is required on the audit event.
4. Declaring whether Finance Centre billing sync is required on toggle.
5. Adding a conditional Setup Wizard step if the extension requires post-provisioning configuration.

No change to the ACP data model or core provisioning engine is required for a standard extension following this pattern, provided the extension does not introduce a new Tenant lifecycle state.

13.6 Acceptance Criteria

The build of the Admin Control Plane is complete when:

- [ ] All canonical objects (Tenant, OnboardingTask, Entitlement, SupportSession, AdminAuditEvent, PmsConnectorRegistration) can be created, read, and updated through the API.
- [ ] The Tenant state machine enforces all transitions defined in §3.2, including go-live smoke test gate.
- [ ] Provisioning is idempotent; re-running any step on an already-provisioned tenant produces no duplicate artefacts and no errors.
- [ ] All integrations in §6 are wired and verified with contract tests.
- [ ] AI boundaries in §7 are enforced; negative tests confirm AI cannot bypass entitlement checks or audit logging.
- [ ] Audit log captures every event in §8; entries with missing mandatory `Reason` fields are rejected at write time.
- [ ] A failed audit write causes the originating action to roll back (rule 6 in §13.2).
- [ ] Access control is enforced per §9; lower-role attempts to toggle regulated extensions are rejected and logged.
- [ ] Support Mode sessions hard-expire at `ExpiresAt` with no code path permitting extension without re-approval.
- [ ] `SupportSessionGranted` SecurityEvent is emitted and successfully written before session access is made available; failure to write blocks session opening.
- [ ] `SupportSessionExpired` SecurityEvent is emitted on both hard expiry and early admin closure; failures surface as operational alerts to SuperAdmin.

- [ ] No cross-tenant data leakage occurs under adversarial negative test conditions.
- [ ] Care Plan Subscriptions Setup Wizard step renders if and only if the entitlement is active.
- [ ] Knowledge, Training & Learning Setup Wizard step renders if and only if the KTL entitlement is active.
- [ ] Medication & Controlled Drugs extension toggle is available only to Provisioning Engineer and SuperAdmin roles.
- [ ] PMS connector registration is rejected without a valid traffic class declaration; rejection is logged.
- [ ] Tenant Health Monitor task-creation events include a deterministic `HealthEventId`; duplicate tasks are not created for the same health event.
- [ ] AI Quality Monitor health signals below the configured severity threshold are surfaced as informational only and do not trigger task creation.
- [ ] All non-functional requirements in §14 are met.

14. Non-Functional Requirements

Performance: Provisioning operations are asynchronous and MUST complete end-to-end (tenant creation through smoke test) within a defined SLA. Synchronous API responses for read operations MUST return within 500 ms at the 95th percentile under normal load. *(Target provisioning SLA not captured in original — needs definition.)*

Availability: The ACP web portal MUST target 99.9% availability during Primoro business hours. Partial degradation (e.g. Tenant Health Monitor alert delivery delayed) is acceptable provided the core provisioning and entitlement surfaces remain operational. Partial failures MUST recover cleanly; no orphaned tenants or entitlements may result from an ACP outage.

Scalability: The ACP MUST support managing at least the full projected tenant count across all regions without per-tenant performance degradation. Multi-tenant data isolation MUST be preserved under concurrent provisioning operations. *(Target tenant count and throughput ceiling not captured in original — needs definition.)*

Security: MFA is enforced for all admin identities. Admin sessions are short-lived. Service-to-service automation uses scoped service tokens. All data in transit between ACP and downstream modules MUST be encrypted using TLS 1.2 or higher. Audit log storage MUST be encrypted at rest. Secrets (PMS tokens, payment gateway credentials, service tokens) MUST be stored in a secrets manager and never in application configuration or source control. *(Specific secrets manager technology not named in original — needs definition.)*

Privacy: The ACP handles Primoro staff identity data and tenant configuration data. It MUST honour applicable data subject rights for admin staff identities. Customer patient data MUST NOT be stored in or transit through ACP systems; the ACP manages tenant configuration only. Audit logs are retained per the organisation's compliance retention policy. *(Specific retention period not captured in original — needs definition.)*

Observability: The ACP MUST export: (a) structured logs for all API requests and audit events; (b) metrics covering provisioning step success/failure rates, SupportSession counts, Entitlement Manager toggle rates, Tenant Health Monitor alert volumes, and PMS connector traffic class validation outcomes; (c) distributed traces for provisioning workflows spanning multiple steps. All exports MUST be available to Primoro platform engineering without requiring direct database access. *(Target observability platform not named in original — needs definition.)*

Accessibility: The ACP web portal is an internal staff tool. It MUST meet WCAG 2.1 AA as a baseline to support Primoro's internal accessibility standards.

15. Open Questions

Provisioning SLA target: What is the defined end-to-end SLA for tenant provisioning (Provisioning → Live)? Referenced in §14 Performance but not defined in the original. *(Surfaced from §14 scaffold.)*

Tenant count and throughput ceiling: What is the maximum projected number of concurrently managed tenants, and at what provisioning concurrency must the system remain performant? *(Surfaced from §14 Scalability.)*

Audit log retention period: What is the mandated retention period for ACP audit logs under SOC 2 and NHS DSPT obligations? *(Surfaced from §7.1 and §14 Privacy.)*

Secrets manager technology: The original names GoCardless and DNA Payments as payment providers and refers to token scoping, but does not name a secrets manager for storing PMS tokens, payment gateway credentials, and service tokens. Which secrets manager is in scope? *(Surfaced from §5.6 PMS Connector Registry and §5.7 Finance Centre.)*

Observability platform: The original requires "inspection-ready audit logs" and mentions health monitoring but does not name the observability platform (metrics, traces, log aggregation). Which platform is in scope? *(Surfaced from §11.4 Acceptance Criteria and §12 Non-Functional Requirements.)*

Tenant Suspended state: The original lifecycle includes a Suspended state implied by billing failure references, but transition rules into and out of Suspended are not defined. What triggers suspension, which role can unsuspend, and what access does the customer retain while suspended? *(Surfaced from §3.2 Tenant State Machine — lifecycle references billing and payment failure without defining Suspended rules.)*

Decommissioning and data retention: The original prohibits deletion of tenants but does not define what happens to tenant data after decommissioning — specifically, whether customer PMS tokens, billing mandates, and telephony numbers are actively revoked or merely flagged. What is the decommissioning procedure? *(Surfaced from §9 Explicit Non-Goals and §11.2 Core Behaviour Rules.)*

ACP-specific SecurityEvent type registry: The original states that ACP MAY define additional admin-specific event types extending the canonical SecurityEvent enumeration. Where is the extension registry maintained, and what is the approval process for adding a new ACP-specific EventType? *(Surfaced from §7.1 Audit Logging, SecurityEvent alignment table. Also relevant to SupportSessionGranted and SupportSessionExpired registration — see §4.10.)*

Lab Manager SLA threshold and accounting export model: Lab Manager §4.1 references practice-level SLA threshold configuration and accounting export triggers via the ACP. The canonical export schema and the set of configurable threshold parameters have not yet been defined. These must be agreed between the Lab Manager and ACP module owners and reflected in a future minor revision of both specs.

AI Quality Monitor event type mapping: The mapping between AI Quality Monitor event types and Tenant Health Monitor `HealthEventType` values (referenced in §4.11) has not yet been formally defined. Primoro platform engineering must produce and maintain this mapping table; its location (e.g. shared schema registry or wiki) is to be confirmed.