Contents

RADIANT Platform Architecture

Enterprise Multi-Tenant AI Platform

Version 4.18.0 | December 2024

A comprehensive technical architecture document for the RADIANT AI orchestration platform

---

Table of Contents

---

1. Platform Overview

1.1 What is RADIANT?

RADIANT (Real-time AI Distribution, Integration, and Automation Network for Tenants) is an enterprise-grade, multi-tenant SaaS platform that provides unified access to 106+ AI models across multiple providers, with intelligent orchestration, cost management, and comprehensive analytics.

1.2 Core Value Proposition

```
                        RADIANT VALUE PROPOSITION



       UNIFIED         INTELLIGENT          COST            ENTERPRISE
       ACCESS         ORCHESTRATION      MANAGEMENT          SECURITY

      106+ AI           49 Multi-          Credits,         SOC2/HIPAA
      Models            AI Patterns        Budgets,         Compliant
      One API           AGI Router         Analytics        Multi-Tenant
```

1.3 Platform Statistics

Metric	Value
AI Models Supported	106+ (50 external + 56 self-hosted)
AI Providers Integrated	15+ (OpenAI, Anthropic, Google, Meta, etc.)
Orchestration Patterns	49 documented patterns
Model Execution Modes	9 (thinking, research, fast, creative, etc.)
Database Migrations	66+ schema migrations
CDK Stacks	14 infrastructure stacks

---

2. System Architecture

2.1 High-Level Architecture

```
                          RADIANT PLATFORM ARCHITECTURE



                           CLIENT LAYER
```

Swift	Admin	Think Tank	SDK &
Deployer	Dashboard	Consumer	API
(macOS)	(Next.js)	App	Clients

API GATEWAY LAYER

Amazon API Gateway

• REST APIs	• WebSocket APIs	• Rate Limiting
• JWT Auth	• API Keys	• Usage Plans

COMPUTE LAYER (Lambda)

Model	Orchestration	Billing	Admin
Router	Engine	Service	Service

Think	Analytics	Learning	Webhook
Tank	Service	Service	Service

DATA LAYER

Aurora	DynamoDB	S3	Redis
PostgreSQL	(Sessions)	(Storage)	(Cache)
(RLS)			

EXTERNAL AI PROVIDERS

OpenAI	Anthropic	Google	Meta	Mistral	+10 more
GPT-4o	Claude	Gemini	Llama		
o1	3.5	2.0	3.1		

2.2 Three-Component Structure

RADIANT consists of three primary deployment components:

```
                        THREE COMPONENTS OF RADIANT


  packages/infrastructure/lambda/shared/services/


                           CORE SERVICES


     model-router.service.ts         Route requests to AI providers
     model-metadata.service.ts       Live model data & capabilities
     orchestration-patterns.service  49 multi-AI workflow patterns
     superior-orchestration.service  Guaranteed superior responses
     learning.service.ts             ML feedback & continuous learning



                          BILLING SERVICES


     billing.service.ts              Credit & subscription management
     cost-management.service.ts      Budget alerts & cost tracking
     usage-analytics.service.ts      Usage metrics & reporting



                         PLATFORM SERVICES


     tenant.service.ts               Multi-tenant management
     auth.service.ts                 Authentication & authorization
     api-key.service.ts              API key lifecycle
     webhook.service.ts              Event notifications
     storage.service.ts              File & artifact storage
```

```
                 THINK TANK SERVICES


thinktank-engine.ts               Multi-step problem solving
thinktank-sessions.ts             Conversation management
collaboration.service.ts          Real-time collaboration
```

---

3. Component Deep Dive

3.1 Model Router Service

The intelligent core that routes AI requests to optimal providers:

```
                       MODEL ROUTER SERVICE



                            Incoming
                            Request




                   Request Validation
                   • API Key Check
                   • Rate Limiting
                   • Tenant Verification





 Model                Budget               Fallback
 Selection            Check                Logic

• Metadata           • Credits            • Primary
• Preferences        • Limits             • Secondary
• Capabilities       • Cost Est.          • Tertiary
```

```
                  Provider Adapter Layer

                OpenAI    Anthropic  ...
                Adapter    Adapter




                Response Processing
                • Token Counting
                • Cost Calculation
                • Usage Recording
                • Analytics Event
```

3.2 Service Architecture

```
                         LAMBDA SERVICES ARCHITECTURE


  packages/infrastructure/lambda/shared/services/


                          CORE SERVICES


    model-router.service.ts          Route requests to AI providers
    model-metadata.service.ts        Live model data & capabilities
    orchestration-patterns.service   49 multi-AI workflow patterns
    superior-orchestration.service   Guaranteed superior responses
    learning.service.ts              ML feedback & continuous learning




                          BILLING SERVICES


    billing.service.ts               Credit & subscription management
    cost-management.service.ts       Budget alerts & cost tracking
    usage-analytics.service.ts       Usage metrics & reporting



                        PLATFORM SERVICES


    tenant.service.ts                Multi-tenant management
    auth.service.ts                  Authentication & authorization
    api-key.service.ts               API key lifecycle
    webhook.service.ts               Event notifications
    storage.service.ts               File & artifact storage



                       THINK TANK SERVICES


    thinktank-engine.ts              Multi-step problem solving
    thinktank-sessions.ts            Conversation management
    collaboration.service.ts         Real-time collaboration
```

---

4. Data Architecture

4.1 Database Schema Overview

```
                     AURORA POSTGRESQL SCHEMA


  66+ Migrations in packages/infrastructure/migrations/


    CORE ENTITIES

    tenants                       Multi-tenant organizations
    users                         User accounts with roles
    api_keys                      API authentication keys
    model_configurations          Per-tenant model settings
    model_metadata                AI model capabilities & pricing
```

BILLING & CREDITS

credit_accounts	Tenant credit balances
credit_transactions	Credit usage history
subscriptions	Plan subscriptions
invoices	Billing invoices
budgets	Spending limits & alerts

ORCHESTRATION

orchestration_methods	Reusable AI method definitions
orchestration_workflows	49 workflow patterns
workflow_method_bindings	Steps linking workflows to methods
orchestration_executions	Execution history & results

THINK TANK

thinktank_sessions	Problem-solving sessions
thinktank_conversations	Conversation threads
thinktank_messages	Individual messages
thinktank_steps	Reasoning steps
thinktank_artifacts	Generated outputs

ANALYTICS & LEARNING

usage_events	API usage events
analytics_aggregates	Pre-computed metrics
learning_interactions	ML training data
model_performance	Model quality tracking

SECURITY

Row-Level Security (RLS) on all tenant tables
SET app.current_tenant_id for automatic filtering
Audit logging on sensitive operations

4.2 Multi-Tenant Data Isolation

```
                  ROW-LEVEL SECURITY (RLS) MODEL


   Request from Tenant A                    Request from Tenant B




    JWT Token                                JWT Token
    tenant_id=A                              tenant_id=B






                     Database Connection
      SET app.current_tenant_id = 'tenant_id_from_jwt';






                      RLS Policy Applied

      CREATE POLICY tenant_isolation ON table_name
      USING (tenant_id = current_setting('app.current_tenant_id'));

      Result: Each tenant ONLY sees their own data


         Tenant A sees:                       Tenant B sees:

      Only Tenant A's                     Only Tenant B's
      - Users                             - Users
      - API Keys                          - API Keys
      - Usage Data                        - Usage Data
      - Conversations                     - Conversations
```

---

5. Security Architecture

5.1 Security Layers

```
                         SECURITY ARCHITECTURE
```

LAYER 1: NETWORK SECURITY

- VPC with private subnets for database
- WAF rules for API Gateway
- CloudFront for DDoS protection
- TLS 1.3 for all connections

LAYER 2: AUTHENTICATION

- Cognito User Pools for user authentication
- JWT tokens with tenant claims
- API Keys with scoped permissions
- MFA support for admin users

LAYER 3: AUTHORIZATION

- Role-based access control (RBAC)
- Permission sets per tenant
- Resource-level policies
- API endpoint authorization

LAYER 4: DATA SECURITY

- Row-Level Security (RLS) in PostgreSQL
- Encryption at rest (AES-256)
- Encryption in transit (TLS)
- KMS for key management
- PHI sanitization for HIPAA compliance

LAYER 5: AUDIT & COMPLIANCE

- CloudTrail for API logging
- Audit tables for data changes
- Compliance reporting dashboard
- SOC2 Type II controls
- HIPAA compliance mode

---

6. Deployment Architecture

6.1 AWS Infrastructure

```
                         AWS DEPLOYMENT ARCHITECTURE



                            REGION: us-east-1


          CloudFront CDN           S3 Static Assets




          API Gateway              Lambda Functions
          (REST + WS)                (Node.js 20)






         Cognito                  Aurora              S3
         User Pools               PostgreSQL          Storage
                                  (Serverless)



         Secrets                  ElastiCache         SQS
         Manager                  (Redis)             Queues
```

6.2 CDK Stack Dependencies

```
                          CDK STACK DEPENDENCIES
```

```
                    NetworkStack
                    (VPC, Subnets)




    AuthStack      DatabaseStack   StorageStack
    (Cognito)       (Aurora)          (S3)




                    AIStack
                 (Model Router)




APIStack          BillingStack       ThinkTankStack


Additional stacks: AnalyticsStack, WebhookStack, ComplianceStack,
                   MonitoringStack, CDNStack, NotificationStack
```

---

7. Integration Points

7.1 External API Integrations

```
                    EXTERNAL INTEGRATIONS


  AI PROVIDERS (15+)


    OpenAI      Anthropic      Google       Meta        Mistral
  GPT-4o, o1 Claude 3.5  Gemini 2.0  Llama 3.1     Large

   Cohere      AI21       Perplexity   DeepSeek     xAI
                            Sonar      R1, Chat     Grok

PAYMENT PROVIDERS

   Stripe      Credit card processing, subscriptions, invoicing

MONITORING & OBSERVABILITY

  CloudWatch    X-Ray        Sentry      Custom Analytics
   (Logs)      (Traces)     (Errors)     Dashboard

NOTIFICATIONS

     SES         SNS        Webhooks     Slack/Teams Integrations
   (Email)     (Push)       (Custom)
```

RADIANT Platform Architecture v4.18.0

Building the future of enterprise AI
