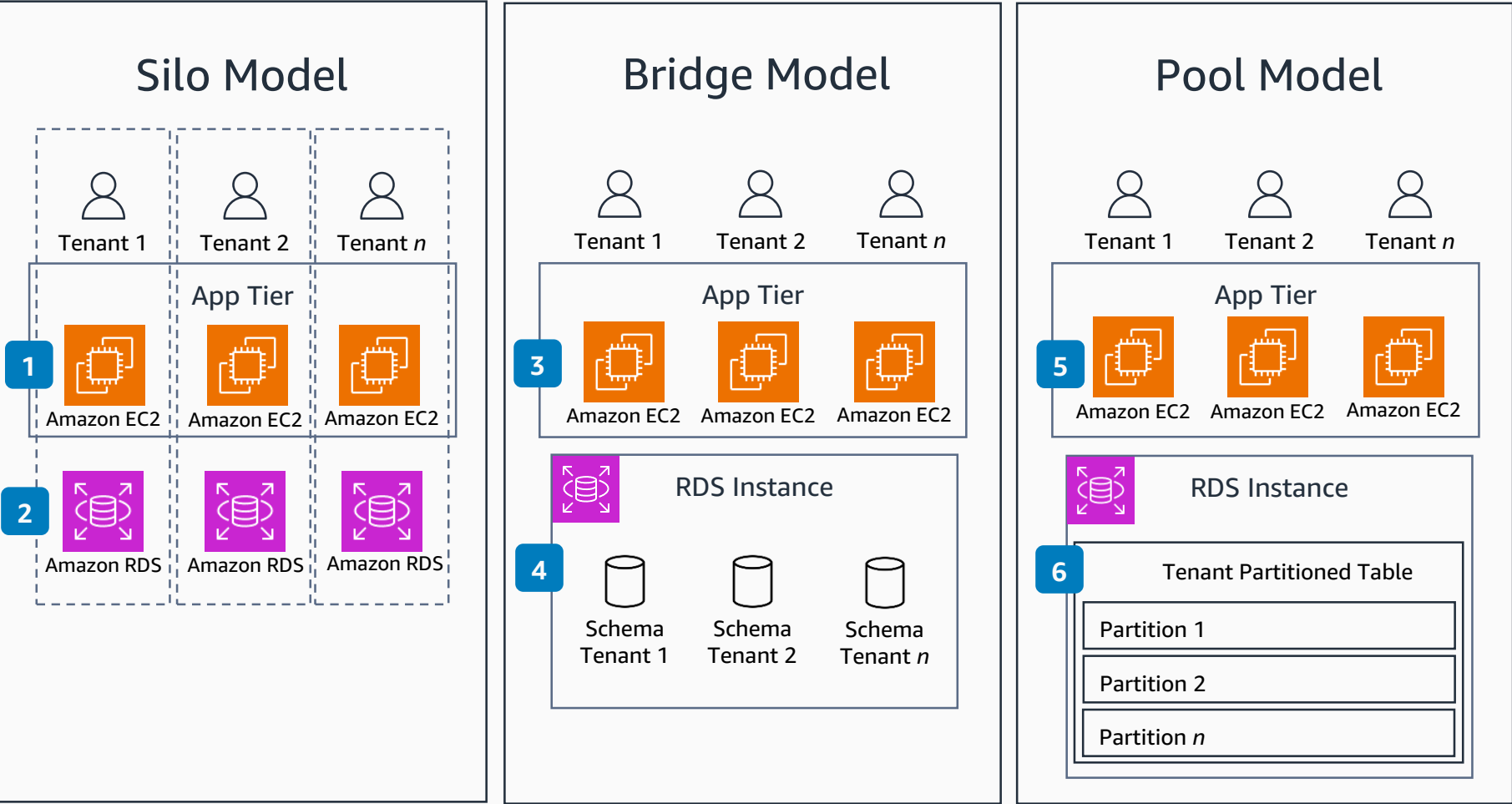# Guidance for Multi-Tenant Architectures on AWS

This architecture shows three approaches for designing multi-tenant SaaS models with Amazon Relational Database Service. In the Silo Model, each tenant is given a dedicated application stack, providing the strictest isolation. In the Bridge Model, tenants share the application stack, but get their own dedicated database schema. In the Pool Model, all resources are shared and isolation is provided by database row-level security.

Silo Model

Tenant 1   Tenant 2   Tenant *n*

App Tier

1
Amazon EC2   Amazon EC2   Amazon EC2

2
Amazon RDS   Amazon RDS   Amazon RDS

Bridge Model

Tenant 1   Tenant 2   Tenant *n*

App Tier

3
Amazon EC2   Amazon EC2   Amazon EC2

RDS Instance

4
Schema Tenant 1   Schema Tenant 2   Schema Tenant *n*

Pool Model

Tenant 1   Tenant 2   Tenant *n*

App Tier

5
Amazon EC2   Amazon EC2   Amazon EC2

RDS Instance

6
Tenant Partitioned Table

| Partition 1 |
| Partition 2 |
| Partition *n* |

Silo Model

1 In the Silo Model, each tenant interacts with an entire application stack dedicated to that tenant. The application stack runs on **Amazon Elastic Compute Cloud (Amazon EC2)** compute instances. Traffic and data do not cross tenant boundaries.

2 Each tenant's data is stored in an **Amazon Relational Database Service (Amazon RDS)** database instance dedicated to that tenant. A tenant's data does not mix with other tenants' data.

Bridge Model

3 In the Bridge Model, tenants share the complete application stack, reducing cost and complexity.

4 Tenants share the same **RDS** instance, but each have a dedicated schema.

Pool Model

5 In the Pool Model, tenants share the entire application stack, just as in the Bridge Model.

6 Tenants share the same database instance and objects. Database tables contain data for multiple tenants. Tenant isolation is provided by row-level security features in the database.

AWS Reference Architecture