Contents

Cato Infrastructure Tier Transitions Runbook

Overview

This runbook covers procedures for managing infrastructure tier transitions between DEV, STAGING, and PRODUCTION tiers.

---

1. Tier Overview

Tier	Monthly Cost	SageMaker Instances	OpenSearch	Use Case
DEV	~$350	0-1 (scale-to-zero)	t3.small (provisioned)	Development, testing
STAGING	~$35K	2-20	r6g.large (provisioned)	Load testing, pre-prod
PRODUCTION	~$750K	50-300	Serverless (50-500 OCUs)	10MM+ users

---

2. Changing Tiers via Admin UI

Step-by-Step

1. Navigate to **System → Infrastructure Tier** in the admin dashboard
2. Review current tier status and costs
3. Enter a reason for the change (minimum 10 characters)
4. Click the target tier card
5. If PRODUCTION tier, confirm the cost warning
6. Wait for transition to complete (5-15 minutes)

Transition Times

Direction	Estimated Time
Scale Up	10-15 minutes
Scale Down	5-10 minutes

---

3. Changing Tiers via API

Request Tier Change

```
curl -X POST https://api.example.com/api/admin/infrastructure/tier/change \
  -H "Authorization: Bearer $ADMIN_TOKEN" \
  -H "Content-Type: application/json" \
  -d '{
    "targetTier": "STAGING",
    "reason": "Load testing for Q1 release"
  }'
```

Response Codes

Code	Status	Action
200	INITIATED	Transition started
200	REQUIRES_CONFIRMATION	Call confirm endpoint
400	REJECTED	Check errors in response

Confirm Tier Change (for PRODUCTION)

```
curl -X POST https://api.example.com/api/admin/infrastructure/tier/confirm \
  -H "Authorization: Bearer $ADMIN_TOKEN" \
  -H "Content-Type: application/json" \
  -d '{
    "confirmationToken": "<token from previous response>"
  }'
```

---

4. Bypassing Cooldown (Emergency Only)

A 24-hour cooldown period is enforced between tier changes. Super admins can bypass this for emergencies.

Requirements

- Super admin role
- Valid emergency reason

Command

```
curl -X POST https://api.example.com/api/admin/infrastructure/tier/bypass-cooldown \
  -H "Authorization: Bearer $SUPER_ADMIN_TOKEN" \
  -H "Content-Type: application/json" \
  -d '{
    "targetTier": "PRODUCTION",
    "reason": "[EMERGENCY] Traffic spike from viral event"
  }'
```

---

5. Monitoring Transition Progress

Via Admin UI

The Infrastructure Tier page shows: - Progress bar during transition - Current step in workflow - Estimated time remaining

Via API

```
curl https://api.example.com/api/admin/infrastructure/tier/transition-status \
  -H "Authorization: Bearer $ADMIN_TOKEN"
```

Via Step Functions Console

1. Go to AWS Step Functions in the console
2. Find state machine: `cato-tier-transition-{environment}`
3. View execution details and current step

---

6. Troubleshooting Failed Transitions

Symptoms

- Transition stuck in "SCALING_UP" or "SCALING_DOWN"
- Error notification received
- Resources partially provisioned

Diagnosis

1. **Check Step Functions execution**

   ```
   aws stepfunctions describe-execution \
     --execution-arn <arn from tier status>
   ```

2. **Check Lambda logs**

   ```
   aws logs filter-log-events \
     --log-group-name /aws/lambda/cato-provision-sagemaker-dev \
     --start-time $(date -d '1 hour ago' +%s000)
   ```

3. **Check resource status**

   ```
   # SageMaker
   aws sagemaker describe-endpoint --endpoint-name cato-shadow-self-<prefix>

   # OpenSearch
   aws opensearch describe-domain --domain-name cato-vectors-<prefix>

   # ElastiCache
   aws elasticache describe-replication-groups --replication-group-id cato-cache-<prefix>
   ```

Resolution

Option 1: Retry Transition Wait for automatic rollback, then retry via admin UI.

Option 2: Manual Reset

```
-- Reset tier state in database
UPDATE cato_infrastructure_tier
SET
  transition_status = 'STABLE',
  target_tier = NULL,
  transition_execution_arn = NULL
WHERE tenant_id = '<tenant-id>';
```

Option 3: Manual Resource Cleanup If resources are partially provisioned:

```
# Delete stuck SageMaker endpoint
aws sagemaker delete-endpoint --endpoint-name cato-shadow-self-<prefix>

# Delete stuck ElastiCache cluster
aws elasticache delete-replication-group \
  --replication-group-id cato-cache-<prefix> \
  --final-snapshot-identifier cato-cache-manual-backup
```

---

7. Rollback Procedures

Automatic Rollback

The Step Functions workflow automatically rolls back on provisioning failure: 1. Detects error during provisioning 2. Calls `rollback-provisioning` Lambda 3. Deletes partially created resources 4. Updates tier state to FAILED 5. Sends alert notification

Manual Rollback

If automatic rollback fails:

1. **Identify partially created resources**

   ```
   aws resourcegroupstaggingapi get-resources \
     --tag-filters Key=TenantId,Values=<tenant-id>
   ```

2. **Delete resources in reverse order**

   - Kinesis streams
   - Neptune instances → clusters
   - ElastiCache clusters
   - OpenSearch domains/collections
   - SageMaker endpoints → configs

3. **Reset database state**

   ```
   UPDATE cato_infrastructure_tier
   SET
     current_tier = '<previous-tier>',
     transition_status = 'STABLE',
     target_tier = NULL
   WHERE tenant_id = '<tenant-id>';
   ```

---

8. Editing Tier Configurations

All tier configurations are admin-editable via the UI.

Via Admin UI

1. Go to **System → Infrastructure Tier**
2. Click "Configure Tiers" tab
3. Click "Edit Configuration" on any tier
4. Modify settings
5. Click "Save Configuration"

Via API

```
curl -X PUT https://api.example.com/api/admin/infrastructure/tier/configs/DEV \
  -H "Authorization: Bearer $ADMIN_TOKEN" \
  -H "Content-Type: application/json" \
  -d '{
    "sagemakerShadowSelfMinInstances": 1,
    "sagemakerShadowSelfMaxInstances": 2,
    "budgetMonthlyCuriosityLimit": 200
  }'
```

Editable Fields

Field	Description	Valid Range
`sagemakerShadowSelfInstanceType`	EC2 instance type	ml.g5.*
`sagemakerShadowSelfMinInstances`	Minimum instances	0-500
`sagemakerShadowSelfMaxInstances`	Maximum instances	1-500
`sagemakerShadowSelfScaleToZero`	Enable scale-to-zero	true/false
`opensearchInstanceType`	OpenSearch instance	t3/r6g.*
`opensearchInstanceCount`	Number of nodes	1-10
`budgetMonthlyCuriosityLimit`	Monthly budget ()|0 − 1000000||‘$budgetDailyExplorationCap$‘|$Dailybudget$()	0-10000

---

9. Cost Verification

After tier transition, verify costs:

Check Estimated Cost

```
curl https://api.example.com/api/admin/infrastructure/tier \
  -H "Authorization: Bearer $ADMIN_TOKEN" | jq '.estimatedMonthlyCost'
```

Check Actual AWS Costs

```
aws ce get-cost-and-usage \
  --time-period Start=$(date -d '-7 days' +%Y-%m-%d),End=$(date +%Y-%m-%d) \
  --granularity DAILY \
  --metrics "BlendedCost" \
  --filter '{
    "Tags": {
      "Key": "Project",
      "Values": ["RADIANT"]
    }
  }'
```

---

10. Emergency Procedures

Runaway Costs

If costs are escalating unexpectedly:

1. **Immediate**: Scale to DEV tier

   ```
   curl -X POST .../tier/bypass-cooldown \
     -d '{"targetTier": "DEV", "reason": "[EMERGENCY] Cost runaway"}'
   ```

2. **Verify**: Check for orphaned resources

   ```
   aws ce get-cost-and-usage-with-resources \
     --time-period Start=$(date +%Y-%m-%d),End=$(date -d '+1 day' +%Y-%m-%d) \
     --granularity DAILY \
     --metrics "BlendedCost" \
     --group-by Type=DIMENSION,Key=RESOURCE_ID
   ```

3. **Cleanup**: Delete any orphaned resources

Production Traffic Spike

If traffic exceeds capacity:

1. **Immediate**: Scale to PRODUCTION tier (bypass cooldown if needed)
2. **Monitor**: Watch SageMaker scaling metrics
3. **Follow-up**: Adjust tier configuration for higher max instances

---

11. Audit and Compliance

View Change History

```
curl https://api.example.com/api/admin/infrastructure/tier/change-history?limit=50 \
  -H "Authorization: Bearer $ADMIN_TOKEN"
```

Database Audit Table

```
SELECT
  from_tier,
  to_tier,
  direction,
  status,
  changed_by,
```

```
  reason,
  started_at,
  completed_at,
  duration_seconds
FROM cato_tier_change_log
WHERE tenant_id = '<tenant-id>'
ORDER BY created_at DESC
LIMIT 20;
```

---

12. Contacts

Role	Contact	Escalation
On-call Engineer	PagerDuty	Tier changes failed
Platform Team	#platform-support	Configuration questions
Finance	finance@example.com	Cost approvals for PRODUCTION

Appendix: Step Functions Workflow States

```
ValidateTransition

       SCALING_UP   ProvisionResources (parallel)
                          ProvisionSageMaker
                          ProvisionOpenSearch
                          ProvisionElastiCache
                          ProvisionNeptune
                          ProvisionKinesis

                        WaitForProvisioning

                        VerifyProvisioning (with retry)

                        UpdateAppConfig

                        TransitionComplete

       SCALING_DOWN   DrainConnections

                         WaitForDrain

                         UpdateAppConfig

                         CleanupResources (parallel)
                            CleanupSageMaker
                            CleanupOpenSearch
```

```
  CleanupElastiCache
  CleanupNeptune

TransitionComplete
```