

# DataLens Alerting Guide

**Document ID:** PRD-DL-025 **Last Updated:** 2024-02-28 **Owner:** DataLens Product Team **Classification:** Public

---

## Overview

DataLens alerting enables you to get notified when your data meets certain conditions. Set up alerts on any query or panel to stay informed about critical metrics.

---

## Alert Types

### Threshold Alerts

Trigger when a metric crosses a defined threshold.

**Examples:** - Revenue drops below \$10,000/day - Error rate exceeds 5% - Response time above 500ms

### Anomaly Alerts

Trigger when a metric deviates from expected patterns.

**Examples:** - Traffic unusually high for time of day - Conversion rate significantly different from historical - User signups trending abnormally

### No Data Alerts

Trigger when expected data stops arriving.

**Examples:** - No orders in past hour - Server stopped reporting metrics - ETL pipeline stalled

---

## Creating Alerts

### From a Panel

1. Open your dashboard
2. Click on the panel title → **Edit**
3. Go to **Alert** tab
4. Click **Create Alert**

### Alert Configuration

**Alert Name:** High Error Rate

**Query:** SELECT COUNT(\*) FROM errors WHERE timestamp > NOW() - INTERVAL '5 min'

#### Conditions:

- **When:** avg()
- **Is Above:** 100
- **For:** 5 minutes

#### Notifications:

- **Channel:** Slack (`#alerts-engineering`)
- **Channel:** Email (`oncall@novatech.com`)

#### Settings:

- Evaluation Interval:** 1 minute
- Pending Period:** 5 minutes
- No Data State:** Alerting
- Error State:** Alerting

### Condition Types

Condition	Description	Example
Is Above	Value exceeds threshold	<code>value &gt; 100</code>
Is Below	Value below threshold	<code>value &lt; 10</code>
Is Outside Range	Value outside bounds	<code>value &lt; 10 OR value &gt; 100</code>
Has No Value	No data returned	Missing metrics
Is Different From	Value changed	<code>value != previous_value</code>

### Aggregation Functions

Function	Description
<code>avg()</code>	Average of values
<code>min()</code>	Minimum value
<code>max()</code>	Maximum value
<code>sum()</code>	Sum of values
<code>count()</code>	Count of values
<code>last()</code>	Most recent value
<code>diff()</code>	Difference from previous
<code>percent_diff()</code>	Percent change from previous

---

## Notification Channels

### Email

1. Go to **Settings** → **Notification Channels**
2. Click **Add Channel** → **Email**
3. Enter email addresses
4. Configure template (optional)
5. Test and save

### Slack

1. Go to **Settings** → **Notification Channels**
2. Click **Add Channel** → **Slack**
3. Click **Add to Slack**
4. Select workspace and channel
5. Test and save

### Slack Message Format:

```

Alert: High Error Rate
Status: Firing
Value: 150 errors
Threshold: > 100
Dashboard: Production Metrics
View: [Link]

```

### PagerDuty

1. Go to **Settings** → **Notification Channels**

2. Click **Add Channel** → **PagerDuty**
3. Enter Integration Key from PagerDuty
4. Configure severity mapping
5. Test and save

## Webhook

Custom integrations via webhook:

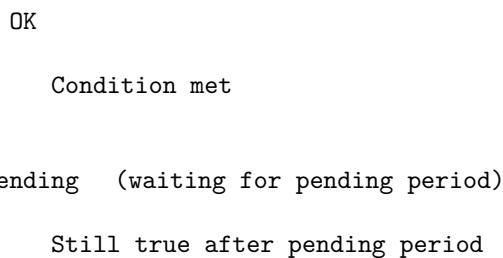
```
{  
  "alert_name": "High Error Rate",  
  "status": "firing",  
  "value": 150,  
  "threshold": 100,  
  "dashboard_url": "https://datalens.novatech.com/d/abc123",  
  "timestamp": "2024-02-28T10:15:00Z"  
}
```

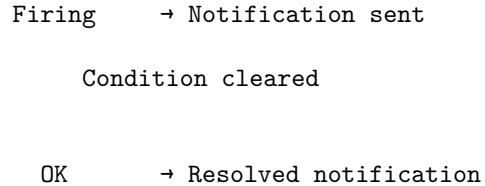
## Microsoft Teams

1. Go to **Settings** → **Notification Channels**
  2. Click **Add Channel** → **Microsoft Teams**
  3. Create incoming webhook in Teams
  4. Paste webhook URL
  5. Test and save
- 

## Alert States

### State Transitions





## State Descriptions

State	Description
OK	Condition not met, all good
Pending	Condition met, waiting for pending period
Firing	Alert triggered, notification sent
No Data	Query returned no data
Error	Query failed to execute

---

## Alert Examples

### Example 1: High Latency Alert

Name: API Latency Alert  
 Query: |  
     SELECT  
         percentile\_cont(0.95) WITHIN GROUP (ORDER BY response\_time) as p95  
     FROM api\_requests  
     WHERE \$\_\_timeFilter(timestamp)

Condition: When p95 Is Above 500 For 5 minutes

Notifications: Slack (#api-alerts)

### Example 2: Revenue Drop Alert

Name: Daily Revenue Alert  
 Query: |  
     SELECT SUM(amount) as revenue  
     FROM orders  
     WHERE created\_at >= CURRENT\_DATE

Condition: When revenue Is Below 10000 For 1 hour

Schedule: Evaluate every 15 minutes  
Notifications: Email (finance@novatech.com)

### Example 3: Error Rate Alert

Name: Error Rate Alert  
Query: |  
SELECT  
 COUNT(\*) FILTER (WHERE status >= 500) \* 100.0 / COUNT(\*) as error\_rate  
FROM http\_requests  
WHERE \$\_\_timeFilter(timestamp)

Condition: When error\_rate Is Above 5 For 3 minutes  
Notifications: PagerDuty (Engineering)

### Example 4: Anomaly Detection

Name: Traffic Anomaly  
Query: |  
SELECT COUNT(\*) as requests  
FROM page\_views  
WHERE \$\_\_timeFilter(timestamp)

Condition: When requests percent\_diff() Is Outside Range -50 to 200  
Notifications: Slack (#traffic-alerts)  
Note: Triggers when traffic is >2x or <0.5x normal

---

## Alert Silencing

### Temporary Silence

Silence alerts during planned events:

1. Go to **Alerting → Silences**
2. Click **New Silence**
3. Configure:
  - **Start/End Time:** When to silence
  - **Matchers:** Which alerts to silence
  - **Comment:** Reason for silencing
4. Click **Create**

## Scheduled Maintenance

Create recurring silences:

```
Name: Weekly Maintenance Window
Schedule: Every Sunday 2:00-4:00 AM UTC
Matchers:
- alertname: ".*" (all alerts)
- severity: "warning"
Comment: Scheduled maintenance window
```

---

## Alert Rules Best Practices

### Reduce Alert Fatigue

1. **Set appropriate thresholds:** Not too sensitive
2. **Use pending periods:** Avoid flapping alerts
3. **Group related alerts:** Don't send duplicates
4. **Prioritize:** Critical vs warning severity
5. **Route appropriately:** Right team, right time

### Effective Alerting

**Alert on symptoms, not causes:** - Good: “High error rate” (symptom) -  
Avoid: “Database connection count high” (cause)

**Make alerts actionable:** - Include context in notifications - Link to relevant  
runbooks - Provide dashboard links

**Set appropriate time windows:** - Short for critical issues (1-5 min) - Longer  
for trends (15-60 min)

---

## Alert Metrics

### Built-in Alert Metrics

DataLens tracks alert performance:

Metric	Description
<code>alerts_firing</code>	Currently firing alerts
<code>alert_state_changes</code>	State transitions
<code>notification_success</code>	Successful notifications
<code>notification_failure</code>	Failed notifications

## Dashboard for Alerts

Create an alerting health dashboard:

```
-- Alerts fired in last 24 hours
SELECT
    alert_name,
    COUNT(*) as fire_count
FROM alert_history
WHERE timestamp > NOW() - INTERVAL '24 hours'
GROUP BY alert_name
ORDER BY fire_count DESC
```

---

## Troubleshooting

### Alert Not Firing

1. **Check query:** Run manually to verify results
2. **Verify threshold:** Ensure condition would trigger
3. **Check evaluation:** Is alert enabled?
4. **Review logs:** Check for query errors

### Too Many Alerts

1. **Increase threshold:** Make less sensitive
2. **Add pending period:** Require sustained condition
3. **Add silence:** For known issues
4. **Group alerts:** Reduce duplicates

### Notifications Not Received

1. **Test channel:** Send test notification
2. **Check configuration:** Verify addresses/tokens

3. **Review delivery:** Check spam/filters
  4. **Verify permissions:** Channel access
- 

## API Reference

### Create Alert

```
curl -X POST https://api.datalens.novatech.com/v1/alerts \
-H "Authorization: Bearer $API_KEY" \
-d '{
  "name": "High Error Rate",
  "query": "SELECT COUNT(*) FROM errors WHERE timestamp > NOW() - INTERVAL \'5 min\'",
  "condition": {
    "type": "threshold",
    "operator": "gt",
    "value": 100
  },
  "notifications": ["slack-channel-id"]
}'
```

### List Alerts

```
curl https://api.datalens.novatech.com/v1/alerts \
-H "Authorization: Bearer $API_KEY"
```

### Silence Alert

```
curl -X POST https://api.datalens.novatech.com/v1/silences \
-H "Authorization: Bearer $API_KEY" \
-d '{
  "matchers": [{"name": "alertname", "value": "High Error Rate"}],
  "startsAt": "2024-02-28T10:00:00Z",
  "endsAt": "2024-02-28T12:00:00Z",
  "comment": "Planned maintenance"
}'
```

---

*Related Documents: Getting Started (PRD-DL-001), Dashboard Creation (PRD-DL-005), Query Language Reference (PRD-DL-010)*