perplexity

```
# <q>provider</q> For Option 1 : Fleet-Wide Pod Monitoring (Single Alarm)  , lets say i h
```

Yes, exactly! With Option 1 (Fleet-Wide Pod Monitoring), you create **just ONE alarm** regardless of whether you have 10 accounts, 100 accounts, or 1000 accounts.[1] [2]

How One Alarm Monitors 1000 Accounts

```
# This creates ONLY 1 alarm in CloudWatch
resource "aws_cloudwatch_metric_alarm" "fleet_wide_single_alarm" {
  provider            = aws.monitoring
  alarm_name          = "all-pods-all-accounts-cpu-high"
  comparison_operator = "GreaterThanThreshold"
  evaluation_periods  = 2
  threshold           = 80

  metric_query {
    id          = "q1"
    return_data = true

    # This query reaches across ALL accounts automatically
    expression = <<-SQL
      SELECT MAX(pod_cpu_utilization)
      FROM SCHEMA("YourCustomNamespace", ClusterName, Namespace, PodName)
      GROUP BY ClusterName, Namespace, PodName, :aws.AccountId
    SQL
  }
}
```

Result:

- **1 alarm resource** in CloudWatch
- Monitors pods across **all 1000 accounts**
- Monitors **thousands or even tens of thousands of pods**
- Each pod is a **contributor** to that single alarm

What "Contributors" Means

When the alarm triggers, CloudWatch shows you which specific contributors caused it:

```
Alarm: all-pods-all-accounts-cpu-high
State: In alarm
Contributors in alarm (showing top 10 of 47):
  1. Account: 111111111111, ClusterName: prod-cluster-1, Namespace: default, PodName: app
```

```
2. Account: 222222222222, ClusterName: prod-cluster-5, Namespace: backend, PodName: api
3. Account: 111111111111, ClusterName: staging-cluster-2, Namespace: default, PodName:
...
```

Limits for Single Fleet-Wide Alarm

However, there are constraints: [3]

Limit	Value	Impact
Max time series per query	500	Alarm can track up to 500 distinct pods simultaneously [3]
Max metrics evaluated per query	10,000	Can query up to 10,000 metrics [3]
Contributors shown in UI	Top 500	Only top 500 contributors displayed [2]

What This Means for Your Scale

If you have **10,000 pods across 1000 accounts**:

✅ **Query succeeds** - can query all 10,000 pods (within 10,000 metric limit)
⚠️ **Only 500 are "tracked" as contributors** - the query returns only 500 time series (you control which 500 with `ORDER BY` and `LIMIT`)

Controlling Which Pods Are Monitored

Use `ORDER BY` and `LIMIT` to focus on the most important pods: [4]

```
resource "aws_cloudwatch_metric_alarm" "top_500_worst_pods" {
  alarm_name = "top-500-cpu-consuming-pods"
  threshold  = 80

  metric_query {
    id = "q1"

    # Automatically monitor the TOP 500 worst performing pods
    expression = <<-SQL
      SELECT MAX(pod_cpu_utilization)
      FROM SCHEMA("YourCustomNamespace", ClusterName, Namespace, PodName)
      GROUP BY ClusterName, Namespace, PodName, :aws.AccountId
      ORDER BY MAX() DESC
      LIMIT 500
    SQL
  }
}
```

Result:

- **1 alarm**
- Monitors **only the 500 pods with highest CPU usage**
- As pods come and go, the alarm **automatically tracks the current top 500**

- Works across **all 1000 accounts**[2] [4]

When You Need Multiple Alarms

You only need multiple fleet-wide alarms if:

Scenario 1: Different Metrics

```
# Alarm 1: CPU monitoring (1 alarm)
resource "aws_cloudwatch_metric_alarm" "fleet_cpu" {
  alarm_name = "fleet-pods-cpu"
  # ... monitors pod_cpu_utilization
}

# Alarm 2: Memory monitoring (1 alarm)
resource "aws_cloudwatch_metric_alarm" "fleet_memory" {
  alarm_name = "fleet-pods-memory"
  # ... monitors pod_memory_utilization
}

# Total: 2 alarms for all 1000 accounts
```

Scenario 2: Different Severity Levels

```
# Alarm 1: Warning threshold (1 alarm)
resource "aws_cloudwatch_metric_alarm" "fleet_warning" {
  alarm_name = "fleet-pods-warning"
  threshold  = 70
  alarm_actions = [aws_sns_topic.warnings.arn]
  # ...
}

# Alarm 2: Critical threshold (1 alarm)
resource "aws_cloudwatch_metric_alarm" "fleet_critical" {
  alarm_name = "fleet-pods-critical"
  threshold  = 90
  alarm_actions = [aws_sns_topic.critical.arn]
  # ...
}

# Total: 2 alarms for all 1000 accounts
```

Scenario 3: Different Environments

```
# Alarm 1: Production pods (1 alarm)
resource "aws_cloudwatch_metric_alarm" "fleet_production" {
  alarm_name = "fleet-pods-production"

  metric_query {
    id = "q1"
    expression = <<-SQL
```

```
      SELECT MAX(pod_cpu_utilization)
      FROM SCHEMA("YourCustomNamespace", ClusterName, Namespace, PodName)
      WHERE Namespace IN ('production', 'prod')
      GROUP BY ClusterName, Namespace, PodName, :aws.AccountId
    SQL
  }
}

# Alarm 2: Staging pods (1 alarm)
resource "aws_cloudwatch_metric_alarm" "fleet_staging" {
  alarm_name = "fleet-pods-staging"

  metric_query {
    id = "q1"
    expression = <<-SQL
      SELECT MAX(pod_cpu_utilization)
      FROM SCHEMA("YourCustomNamespace", ClusterName, Namespace, PodName)
      WHERE Namespace IN ('staging', 'stage')
      GROUP BY ClusterName, Namespace, PodName, :aws.AccountId
    SQL
  }
}

# Total: 2 alarms for all 1000 accounts
```

Terraform Count Verification

To make it crystal clear, let's show the Terraform plan output:

```
$ terraform plan

Terraform will perform the following actions:

  # aws_cloudwatch_metric_alarm.fleet_wide_single_alarm will be created
  + resource "aws_cloudwatch_metric_alarm" "fleet_wide_single_alarm" {
      + alarm_name          = "all-pods-all-accounts-cpu-high"
      + comparison_operator = "GreaterThanThreshold"
      ...
    }

Plan: 1 to add, 0 to change, 0 to destroy.
```

Only 1 resource gets created, no matter how many accounts or pods.[1]

Cost Implications

One alarm for 1000 accounts is extremely cost-effective:

- **Metrics Insights alarm**: First 10 alarms free, then $0.10/alarm/month[5]
- **Your cost**: $0 (assuming you have <10 total Metrics Insights alarms)
- **Alternative (individual alarms)**: If you created 10,000 individual alarms at $0.10 each = $1,000/month

Summary

Option 1: Fleet-Wide Pod Monitoring

- **Number of accounts**: 1000
- **Number of pods**: 10,000+
- **Number of alarms created**: **1** ✅
- **Pods tracked per alarm**: Up to 500 contributors
- **Terraform resources**: 1

This is the power of CloudWatch Metrics Insights with GROUP BY - **one alarm can monitor an entire fleet** across any number of accounts.[2] [3] [1]

⁂

1. https://pubudu.dev/posts/cloudwatch-multi-metric-alarms/
2. https://www.amazonaws.cn/en/new/2025/amazon-cloudwatch-query-alarms-support-monitoring-metrics-individually/
3. https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/cloudwatch-metrics-insights-limits.html
4. https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/cloudwatch-metrics-insights-querylanguage.html
5. https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/cloudwatch_limits.html