

Inventory Metric

Endpoints Summary

Method	Path	Swagger
GET	/inventory_metric/	Swagger ↗
GET	/inventory_metric/{id}/	Swagger ↗

“ The Inventory Metric API provides endpoints for retrieving and managing inventory metrics within the Zequence Control system. These endpoints allow you to access detailed information about various measurement parameters used to track inventory performance, resource utilization, and operational statistics across your infrastructure.

Base URL: `https://control.zequence.com/api/v1`

Authentication: All endpoints require a Bearer token:

```
Authorization: Bearer <your-api-token>
```

Overview

The Inventory Metric API is designed to help you access and manage the various metrics that track your inventory system's performance and status. Metrics are essential measurement points that provide insights into resource utilization, operational efficiency, and system health across your infrastructure.

These endpoints are commonly used for:

- **Monitoring and Analytics:** Retrieve metrics data for dashboards, reports, and monitoring systems
- **Performance Tracking:** Access specific metrics to analyze trends and identify optimization opportunities

- **Integration:** Connect inventory metrics with external monitoring tools and business intelligence platforms
- **Operational Visibility:** Get real-time and historical data about inventory operations

The API supports three types of metrics: Absolute (point-in-time values), Counter (cumulative values that increase over time), and Delta (change values between measurements). Each metric can be organized into groups and linked to specific parameters for structured data organization.

Endpoints

GET /inventory_metric/

Description: Retrieves a paginated list of all inventory metrics available in your system. This endpoint is essential for discovering available metrics, building metric catalogs, and implementing metric selection interfaces in applications.

Use Cases:

- Building dashboards that allow users to select from available metrics
- Creating monitoring configurations by discovering metric names and types
- Auditing and documenting all metrics tracked in your inventory system
- Filtering metrics by name to find specific measurement categories

Full URL Example:

```
https://control.zequence.com/api/v1/inventory_metric/?name=cpu_usage&limit=20&offset=0
```

Parameters:

Parameter	Type	In	Required	Description
name	string	query	No	Filter results by metric name. Supports partial matching for finding metrics containing specific keywords
limit	integer	query	No	Number of results to return per page. Default is typically 20, maximum varies by system configuration

Parameter	Type	In	Required	Description
offset	integer	query	No	The initial index from which to return results. Used for pagination through large metric lists

cURL Example:

```
curl -X GET
"https://control.zequence.com/api/v1/inventory_metric/?name=storage&limit=10&offset=0" \
-H "Authorization: Bearer YOUR_API_TOKEN" \
-H "Content-Type: application/json"
```

Example Response:

```
{
  "count": 45,
  "next": "https://control.zequence.com/api/v1/inventory_metric/?limit=10&offset=10",
  "previous": null,
  "results": [
    {
      "id": 101,
      "name": "storage_utilization_percent",
      "group": 5,
      "description": "Percentage of storage space currently utilized across all storage
devices",
      "type": "a",
      "parameter": 1001
    },
    {
      "id": 102,
      "name": "storage_capacity_gb",
      "group": 5,
      "description": "Total storage capacity in gigabytes available in the inventory",
      "type": "a",
      "parameter": 1002
    },
    {
      "id": 103,
      "name": "storage_requests_total",
```

```
    "group": 5,
    "description": "Cumulative count of storage allocation requests processed",
    "type": "c",
    "parameter": 1003
  }
]
}
```

Response Codes:

Status	Description
200	Success - Returns paginated list of inventory metrics
401	Unauthorized - Invalid or missing Bearer token
403	Forbidden - Insufficient permissions to access metrics
500	Internal Server Error - System error occurred

GET /inventory_metric/{id}/

Description: Retrieves detailed information about a specific inventory metric by its unique identifier. This endpoint provides complete metric configuration details including its type, description, associated group, and parameter relationships.

Use Cases:

- Getting complete details for a specific metric to display in monitoring interfaces
- Validating metric configuration before using it in calculations or reports
- Understanding metric relationships and groupings for proper data interpretation
- Retrieving metric metadata for documentation and system integration purposes

Full URL Example:

```
https://control.zequence.com/api/v1/inventory_metric/101/
```

Parameters:

Parameter	Type	In	Required	Description
id	integer	path	Yes	Unique identifier of the inventory metric to retrieve

Parameter	Type	In	Required	Description
name	string	query	No	Additional filter by metric name (typically not needed when using specific ID)

cURL Example:

```
curl -X GET "https://control.zequenze.com/api/v1/inventory_metric/101/" \
-H "Authorization: Bearer YOUR_API_TOKEN" \
-H "Content-Type: application/json"
```

Example Response:

```
{
  "id": 101,
  "name": "cpu_utilization_percent",
  "group": 3,
  "description": "Average CPU utilization percentage across all compute resources in inventory",
  "type": "a",
  "parameter": 2001
}
```

Response Codes:

Status	Description
200	Success - Returns the requested inventory metric details
401	Unauthorized - Invalid or missing Bearer token
403	Forbidden - Insufficient permissions to access this metric
404	Not Found - Metric with specified ID does not exist
500	Internal Server Error - System error occurred

Common Use Cases

Use Case 1: Building a Metrics Dashboard

Retrieve all available metrics using the list endpoint, then fetch specific metric details to populate dashboard configuration options. Filter by name to find metrics related to specific categories like "cpu", "memory", or "network".

Use Case 2: Monitoring System Integration

Use both endpoints to discover available metrics and their types (Absolute, Counter, Delta) to properly configure external monitoring tools that need to understand how to aggregate and display the data correctly.

Use Case 3: Metric Validation and Documentation

Fetch individual metrics by ID to validate their current configuration, understand their parameter relationships, and generate documentation about what metrics are available in your inventory system.

Use Case 4: Performance Analysis Setup

Query metrics by name patterns to find all performance-related metrics, then use the detailed endpoint to understand their types and descriptions for setting up automated performance analysis workflows.

Use Case 5: Audit and Compliance Reporting

Retrieve comprehensive lists of all tracked metrics to ensure compliance with monitoring requirements and to document what operational data is being collected from your inventory systems.

Best Practices

- **Use Pagination Effectively:** When retrieving large numbers of metrics, use appropriate limit values (typically 20-100) to balance performance with data completeness
- **Cache Metric Metadata:** Metric definitions change infrequently, so cache the results to reduce API calls when building applications that reference metrics repeatedly
- **Filter by Name Strategically:** Use the name filter to find metrics efficiently rather than retrieving all metrics and filtering client-side
- **Understand Metric Types:** Pay attention to the type field (a/c/d) as it affects how you should interpret and aggregate metric values in your applications

- **Handle Group Relationships:** Use the group field to organize related metrics together in user interfaces and reports
 - **Implement Proper Error Handling:** Always check for 404 responses when requesting specific metric IDs, as metrics may be removed or reorganized
 - **Monitor API Rate Limits:** Be mindful of request frequency, especially when iterating through multiple individual metric endpoints
-

Revision #4

Created 2026-02-04 05:10:28 UTC by ipena@zequenze.com

Updated 2026-02-11 03:06:31 UTC by ipena@zequenze.com