

Hitachi Command Suite

Tuning Manager API Reference Guide

FASTFIND LINKS

- Document Organization
- Product Version
- Getting Help
- Contents

© 2014, 2015, Hitachi, Ltd. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, or stored in a database or retrieval system for any purpose without the express written permission of Hitachi, Ltd.

Hitachi, Ltd., reserves the right to make changes to this document at any time without notice and assumes no responsibility for its use. This document contains the most current information available at the time of publication. When new or revised information becomes available, this entire document will be updated and distributed to all registered users.

Some of the features described in this document might not be currently available. Refer to the most recent product announcement for information about feature and product availability, or contact Hitachi Data Systems Corporation at <https://portal.hds.com>.

Notice: Hitachi, Ltd., products and services can be ordered only under the terms and conditions of the applicable Hitachi Data Systems Corporation agreements. The use of Hitachi, Ltd., products is governed by the terms of your agreements with Hitachi Data Systems Corporation.

Hitachi is a registered trademark of Hitachi, Ltd., in the United States and other countries. Hitachi Data Systems is a registered trademark and service mark of Hitachi, Ltd., in the United States and other countries.

Archivas, Essential NAS Platform, HiCommand, Hi-Track, ShadowImage, Tagmaserve, Tagmasoft, Tagmasolve, Tagmastore, TrueCopy, Universal Star Network, and Universal Storage Platform are registered trademarks of Hitachi Data Systems.

AIX, AS/400, DB2, Domino, DS6000, DS8000, Enterprise Storage Server, ESCON, FICON, FlashCopy, IBM, Lotus, MVS, OS/390, RS/6000, S/390, System z9, System z10, Tivoli, VM/ESA, z/OS, z9, z10, zSeries, z/VM, and z/VSE are registered trademarks or trademarks of International Business Machines Corporation.

All other trademarks, service marks, and company names in this document or web site are properties of their respective owners.

Microsoft product screen shots are reprinted with permission from Microsoft Corporation.

Notice on Export Controls. The technical data and technology inherent in this Document may be subject to U.S. export control laws, including the U.S. Export Administration Act and its associated regulations, and may be subject to export or import regulations in other countries. Reader agrees to comply strictly with all such regulations and acknowledges that Reader has the responsibility to obtain licenses to export, re-export, or import the Document and any Compliant Products.

EXPORT CONTROLS - Licensee will comply fully with all applicable export laws and regulations of the United States and other countries, and Licensee shall not export, or allow the export or re-export of, the API in violation of any such laws or regulations. By downloading or using the API, Licensee agrees to the foregoing and represents and warrants that Licensee is not located in, under the control of, or a national or resident of any embargoed or restricted country.



Contents

Preface.....	v
Intended audience.....	vi
Product version.....	vi
Release notes.....	vi
Document organization.....	vi
Related documents.....	vi
Document conventions.....	vii
Conventions for storage capacity values.....	viii
Accessing product documentation.....	viii
Getting help.....	ix
Comments.....	ix
1 Overview.....	1-1
Overview of the Tuning Manager API.....	1-2
Information that can be obtained via http(s).....	1-3
Information that can be updated via http(s).....	1-4
Conditions for using the API to access performance data.....	1-4
General procedure for installation and setup.....	1-4
General procedure for installation and setup (When Agents operate in a non-cluster system).....	1-5
General procedure for installation and setup (When Agents operate in a cluster system).....	1-6
2 Specification of the API.....	2-1
API common specifications.....	2-2
Supported HTTP methods	2-4
Tuning Manager API version compatibility.....	2-4
Request message.....	2-4
Request line.....	2-4
Request headers.....	2-6
Security and authentication.....	2-8
Transmission in compressed format.....	2-8
Common objects.....	2-9
Collection object.....	2-9
Error object.....	2-9

Action object.....	2-10
PrimitiveValue object.....	2-10
Version object.....	2-11
Job object.....	2-11
3 List of resources.....	3-1
List of Tuning Manager API resources.....	3-2
API functions.....	3-6
Obtaining the detailed version information of the API.....	3-6
Obtaining a list of Agents administrative actions.....	3-8
Obtaining the Agents list parameter when the manual refresh process is executed.....	3-9
Manually refreshing the Agents list.....	3-11
Obtaining a list of Agent for RAID instances.....	3-15
Obtaining performance data.....	3-18
Obtaining metadata.....	3-28
General procedure for obtaining performance data.....	3-30
4 API Messages.....	4-1
Message Format.....	4-2
Format of Output Messages.....	4-2
Message Description Format.....	4-2
List of Output Destinations for Messages.....	4-2
Messages	4-3
A Status Codes.....	A-1
Status codes.....	A-2
B Specification differences depending on the Performance database being used.....	B-1
Specification differences.....	B-2

Acronyms and Abbreviations



Preface

This manual is a reference for using APIs provided by the Tuning Manager series (Tuning Manager APIs).

This preface includes the following information:

- [Intended audience](#)
- [Product version](#)
- [Release notes](#)
- [Document organization](#)
- [Related documents](#)
- [Document conventions](#)
- [Conventions for storage capacity values](#)
- [Accessing product documentation](#)
- [Getting help](#)
- [Comments](#)

Intended audience

- Users who want to use an API to obtain the information of the monitoring target for Agents via http(s)
- Users who have basic knowledge of Representational State Transfer (REST) architecture

Product version

This document revision applies to Tuning Manager v8.1.3 or later.

Release notes

Read the release notes before installing and using this product. They may contain requirements or restrictions that are not fully described in this document or updates or corrections to this document.

Document organization

The following table provides an overview of the contents and organization of this document. Click the chapter title in the left column to go to that chapter. The first page of each chapter provides links to the sections in that chapter.

Chapter	Description
Chapter 1, Overview on page 1-1	This chapter describes the overview of the Tuning Manager API.
Chapter 2, Specification of the API on page 2-1	This chapter describes the specifications of the Tuning Manager API.
Chapter 3, List of resources on page 3-1	This chapter lists the Tuning Manager API resources, and describes the functions of each URI. Also, this chapter gives examples of the general procedure for obtaining performance data by using the API.
Chapter 4, API Messages on page 4-1	This chapter describes the Tuning Manager API message format and lists the API messages.

Related documents

The following related Hitachi Command Suite documents are available on the documentation CD:

- *Hitachi Command Suite Tuning Manager Agent Administration Guide*, MK-92HC013
- *Hitachi Command Suite Tuning Manager Server Administration Guide*, MK-92HC021


- *Hitachi Command Suite Tuning Manager User Guide*, MK-92HC022
- *Hitachi Command Suite Tuning Manager Hardware Reports Reference*, MK-95HC111
- *Hitachi Command Suite Tuning Manager Operating System Reports Reference*, MK-95HC112
- *Hitachi Command Suite Tuning Manager Application Reports Reference*, MK-95HC113
- *Hitachi Command Suite Tuning Manager Messages*, MK-95HC114
- *Hitachi Command Suite Tuning Manager CLI Reference Guide*, MK-96HC119
- *Hitachi Command Suite Tuning Manager Installation Guide*, MK-96HC141




Document conventions

This document uses the following typographic conventions:

Convention	Description
Bold	Indicates text on a window, other than the window title, including menus, menu options, buttons, fields, and labels. Example: Click OK .
<i>Italic</i>	Indicates a variable, which is a placeholder for actual text provided by the user or system. Example: <i>copy source-file target-file</i> <i>Note:</i> Angled brackets (< >) are also used to indicate variables.
Monospace	Indicates text that is displayed on screen or entered by the user. Example: # pairdisplay -g oradb
< > angled brackets	Indicates a variable, which is a placeholder for actual text provided by the user or system. Example: # pairdisplay -g <group> <i>Note:</i> Italic font is also used to indicate variables.
[] square brackets	Indicates optional values. Example: [a b] indicates that you can choose a, b, or nothing.
{ } braces	Indicates required or expected values. Example: { a b } indicates that you must choose either a or b.
vertical bar	Indicates that you have a choice between two or more options or arguments. Examples: [a b] indicates that you can choose a, b, or nothing. { a b } indicates that you must choose either a or b.

This document uses the following icons to draw attention to information:

Icon	Label	Description
	Note	Calls attention to important or additional information.

Icon	Label	Description
	Tip	Provides helpful information, guidelines, or suggestions for performing tasks more effectively.
	Caution	Warns the user of adverse conditions or consequences (for example, disruptive operations).
	WARNING	Warns the user of severe conditions or consequences (for example, destructive operations).

Conventions for storage capacity values

Physical storage capacity values (for example, disk drive capacity) are calculated based on the following values:

Physical capacity unit	Value
1 kilobyte (KB)	1,000 (10^3) bytes
1 megabyte (MB)	1,000 KB or $1,000^2$ bytes
1 gigabyte (GB)	1,000 MB or $1,000^3$ bytes
1 terabyte (TB)	1,000 GB or $1,000^4$ bytes
1 petabyte (PB)	1,000 TB or $1,000^5$ bytes
1 exabyte (EB)	1,000 PB or $1,000^6$ bytes

Logical storage capacity values (for example, logical device capacity) are calculated based on the following values:

Logical capacity unit	Value
1 block	512 bytes
1 KB	1,024 (2^{10}) bytes
1 MB	1,024 KB or $1,024^2$ bytes
1 GB	1,024 MB or $1,024^3$ bytes
1 TB	1,024 GB or $1,024^4$ bytes
1 PB	1,024 TB or $1,024^5$ bytes
1 EB	1,024 PB or $1,024^6$ bytes

Accessing product documentation

The Tuning Manager user documentation is available on the Hitachi Data Systems Portal: <https://portal.hds.com>. Check this site for the most

current documentation, including important updates that may have been made after the release of the product.

Getting help

Hitachi Data Systems Support Portal is the destination for technical support of your current or previously-sold storage systems, midrange and enterprise servers, and combined solution offerings. The Hitachi Data Systems customer support staff is available 24 hours a day, seven days a week. If you need technical support, log on to the Hitachi Data Systems Support Portal for contact information: <https://portal.hds.com>.

Hitachi Data Systems Community is a new global online community for HDS customers, partners, independent software vendors, employees, and prospects. It is an open discussion among these groups about the HDS portfolio of products and services. It is the destination to get answers, discover insights, and make connections. The HDS Community complements our existing Support Portal and support services by providing an area where you can get answers to non-critical issues and questions. **Join the conversation today!** Go to community.hds.com, register, and complete your profile.

Comments

Please send us your comments on this document: doc.comments@hds.com. Include the document title and number, including the revision level (for example, -07), and refer to specific sections and paragraphs whenever possible. All comments become the property of Hitachi Data Systems Corporation.

Thank you!

Overview

This chapter provides an overview of the API (Tuning Manager API) that Tuning Manager provides.

- [Overview of the Tuning Manager API](#)
- [Information that can be obtained via http\(s\)](#)
- [Information that can be updated via http\(s\)](#)
- [Conditions for using the API to access performance data](#)
- [General procedure for installation and setup](#)

Overview of the Tuning Manager API

The Tuning Manager series allow you to use an API to obtain data via http(s), such as monitoring target information collected by Agents. The API is called a Tuning Manager API.

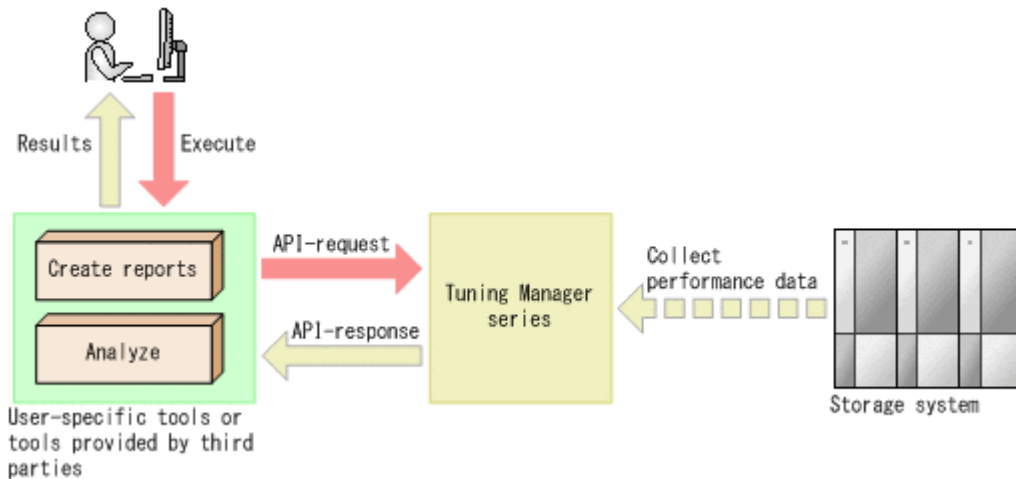


Figure 1-1 Overview of the Tuning Manager API

The advantages of using the API are as follows:

- You can remotely obtain information over a network.
- You can obtain a large amount of performance data quickly.
- You can obtain only the information you need, and create unique reports.
- By using user-specific tools or tools provided by third parties, you can analyze the performance data obtained by the API.

The Tuning Manager API follows the Representational State Transfer (REST) architecture style, and therefore has the following advantages:

- Independent of platforms and languages
- Conforms to the HTTP standard interface (among the GET, PUT, POST, and DELETE methods, the Tuning Manager API can use the GET and POST methods.)
- The API can use HTTP security tools.
- The API returns standard HTTP status codes, allowing for prompt action if an error occurs.
- Each resource has an Uniform Resource Identifier (URI), providing easy resource access. The relationship between resources is also easy to understand.

You can use the API to obtain performance data collected by the following Agents:

- Agent for RAID

You can use the API to obtain performance data corresponding to the minimum data collection interval. The following shows examples of the default data collection intervals when using Agent for RAID:

- For port, cache, or processor: 1 minute
- For disk resources such as a parity group and LDEV: 5 minutes

For details on the system configuration and settings for using the API, see the following manuals.

- System configuration for obtaining data by using the API:
Tuning Manager User Guide
- Settings set by the Tuning Manager server:
Tuning Manager Server Administration Guide and *Tuning Manager Installation Guide*
- Settings set by Agents:
Tuning Manager Agent Administration Guide

Information that can be obtained via http(s)

The following information can be obtained by using API:

- Performance data collected by Agents
You can obtain the performance data that Agents collect from the monitoring targets.
You can obtain the log data collected in each collection interval, as well as summarized data, such as hourly or daily data. You cannot obtain real-time data.
In addition, the default retention period differs depending on the data type of the log data and each summarized data item. For details about the default value for each data item, see the manual *Hitachi Command Suite System Requirements*.
- Note
For the information sent from the agents operated on the Store database, the default retention period of the log data is 48 hours. However, the summarized data cannot be obtained.
For details on how to change the retention period, see the *Tuning Manager Agent Administration Guide*.
- API configuration information
You can obtain information including API version and the series version of Tuning Manager.
- Agents list
You can obtain list of instance information and monitoring target information for Agents.
- Metadata of the Agents information and metadata of resources used for the API
You can obtain metadata (such as attributes and types) of data that was obtained by using the API.

Information that can be updated via http(s)

This API can update Agents list.

Note that the information in the Agents list is regularly and automatically updated. The update interval can be modified.

For details about automatic updates, see the explanations about refreshing agent information in the manual *Tuning Manager Server Administration Guide*.

Conditions for using the API to access performance data

The API can be used to access performance data if the following conditions are met:

- A Tuning Manager server license is registered.
- The user who wants to access the data has the View permission or higher for the Tuning Manager server.
- The system is operated by using Hybrid Store^{#1}.
- The use of Tuning Manager API is enabled.
- In Agents, there is at least one instance of the performance data that the user wants to obtain by using the API.
- The performance data that the user wants to obtain by using the API is regularly collected by Agents.^{#2}
- Records are set to be output to Hybrid Store^{#2}. (By default, the records collected at regular intervals are output to Hybrid Store.)

#1

Performance data can be accessed also when the system is operated by using Store database as well as when use of the Tuning Manager API is enabled.

#2

When the system is operated by using Store database as well as when use of the Tuning Manager API is enabled, the records must be set to be output to the performance data file when a Store database is used. By default, the records collected at regular intervals are output to the performance data file.

General procedure for installation and setup

This section explains the general procedure for installation and setup required for using Tuning Manager API.

If Agents that enable the Tuning Manager API operate in a non-clustered system, see [General procedure for installation and setup \(When Agents operate in a non-cluster system\)](#) on page 1-5. If they operate in a cluster

system, see [General procedure for installation and setup \(When Agents operate in a cluster system\)](#) on page 1-6.

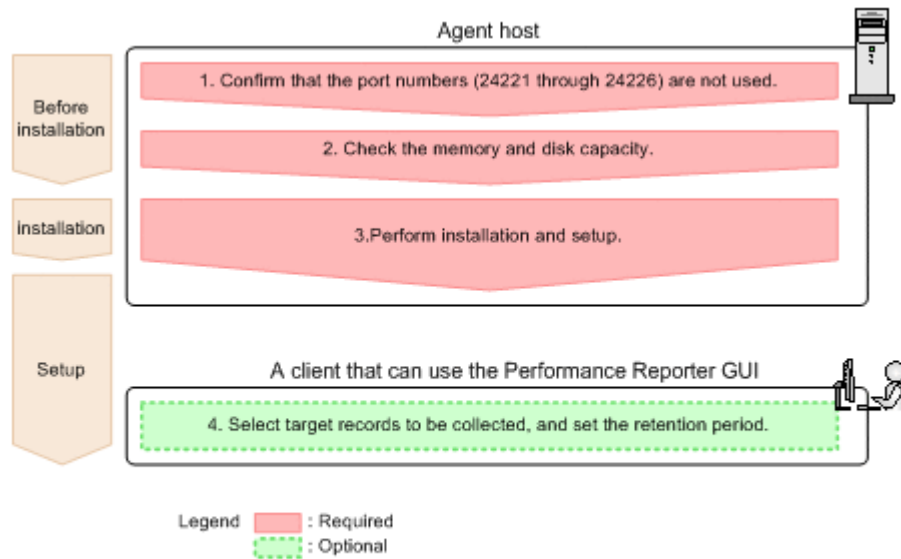
Note that this section only explains the tasks that are required when using the Tuning Manager API.

For details on the normal installation and setup, see *Tuning Manager Installation Guide*.

General procedure for installation and setup (When Agents operate in a non-cluster system)

This section explains the general procedure for installation and setup required for using Tuning Manager API when Agents that enable the Tuning Manager API operate in a non-clustered system.

Note that this section only explains the tasks that are required when using the Tuning Manager API.



Note: For details on each task shown in this figure, see the reference information indicated in the table below.

Figure 1-2 General procedure for installation and setup required for using the Tuning Manager API (When Agents operate in a non-cluster system)

Table 1-1 References for tasks in the flow chart (When Agents operate in a non-cluster system)

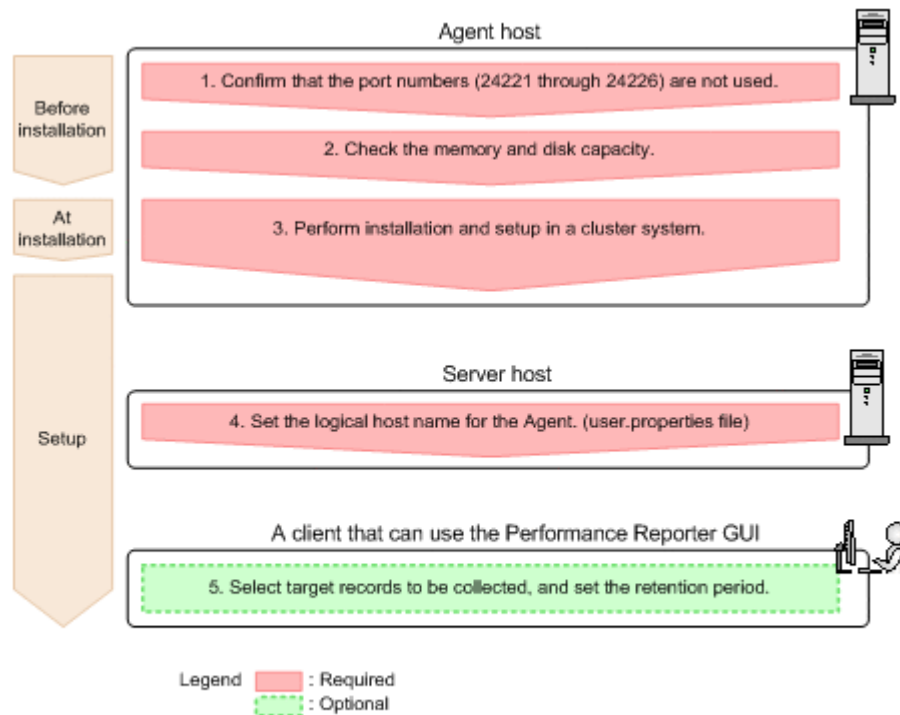
Number and task in the flow chart		Reference manual
1	Confirm that the port numbers (24221 through 24226) are not used	<p><i>Tuning Manager Agent Administration Guide</i></p> <p>If a port number from 24221 through 24226 is used, see the description about changing the port number when a user uses the Tuning Manager API. Then, change the port number.</p>

Number and task in the flow chart			Reference manual
2	Check the memory and disk capacity	Required	<i>Hitachi Command Suite System Requirements</i>
			See the description of system estimates.
3	Perform installation and setup	Required	<i>Tuning Manager Installation Guide</i>
			See the description of the installation and setup procedures.
4	Select target records to be collected, and set the retention period	Optional	<i>Tuning Manager Agent Administration Guide</i>
			See the description of the settings related to output to the performance data.

General procedure for installation and setup (When Agents operate in a cluster system)

This section explains the general procedure for installation and setup required for using Tuning Manager API when Agents that enable the Tuning Manager API operate in a cluster system.

Note that this section only explains the tasks that are required when using the Tuning Manager API.



Note: For details on each task shown in this figure, see the reference information indicated in the table below.

Figure 1-3 General procedure for installation and setup required for using the Tuning Manager API (When Agents operate in a cluster system)

Table 1-2 References for tasks in the flow chart (When Agents operate in a cluster system)

Number and task in the flow chart			Reference manual
1	Confirm that the port numbers (24221 through 24226) are not used	Required	<i>Tuning Manager Agent Administration Guide</i>
			If a port number from 24221 through 24226 is used, see the description about changing the port number when a user uses the Tuning Manager API. Then, change the port number.
2	Check the memory and disk capacity	Required	<i>Hitachi Command Suite System Requirements</i>
			See the description of system estimates.
3	Perform installation and setup in a cluster system	Required	<i>Tuning Manager Installation Guide</i>
			See the description of the installation and setup procedures in a cluster system.
4	Set the logical host name of an Agent (<code>user.properties</code> file)	Required	<i>Tuning Manager Server Administration Guide</i>
			See the description of the settings when the host name of the Agent for which use of the Tuning Manager API is enabled is changed.
5	Select target records to be collected, and set the retention period	Optional	<i>Tuning Manager Agent Administration Guide</i>
			See the description of the settings related to output to the performance data.

Specification of the API

This chapter explains the specifications of the Tuning Manager API.

- [API common specifications](#)
- [Supported HTTP methods](#)
- [Tuning Manager API version compatibility](#)
- [Request message](#)
- [Security and authentication](#)
- [Transmission in compressed format](#)
- [Common objects](#)

API common specifications

This section describes the common specifications of the API for access data via http.

Use URI to specify a resource to be handled by API. In the following URI format, the base URI is from <schema> to <version>, and the relative URI is from <domain>. Specify resources to the relative URI from <domain>.

The followings describes URI formats, and [Table 2-1 URI elements on page 2-2](#) describes URI elements.

Format:

```
scheme://host:port/application/version[/domain[/object-type]][?query-string]
```

or

```
scheme://host:port/application/version/domain/object-type[/actions[/action-identifier[/invoke]]]
```

Table 2-1 URI elements

Element	Description
<i>scheme</i>	The protocol used for communication. Both <code>http</code> and <code>https</code> are supported.
<i>host</i>	The host name or IP address of the Tuning Manager server host that the API client is to access.
<i>port</i>	The port number of the Tuning Manager server host that the API client is to access. The default is 22015 if the protocol is HTTP. The default is 22016 if the protocol is HTTPS.
<i>application</i>	The application name. Specify <code>TuningManager</code> .
<i>version</i>	The version of the Tuning Manager API. This element becomes the base URI of the API service. The version is indicated in the format <code>vx</code> (x indicates the major version of the API). If the API version is <code>1.y</code> (y indicates the minor version of the API). For example, <code>1.y</code> can refer to <code>1.0</code> , <code>1.1</code> , or <code>1.2.</code>), <code><version></code> is displayed as <code>v1</code> . If the API version is <code>2.y</code> (for example, <code>2.0</code> , <code>2.1</code> , or <code>2.2</code>), <code><version></code> is displayed as <code>v2</code> . For details about the correspondence between versions of the Tuning Manager series and of the API, see Tuning Manager API version compatibility on page 2-4 .
<i>domain</i>	The resource category. # The following 4 domains are available: <ul style="list-style-type: none"><code>configuration</code> Detailed API information<code>services</code>

Element	Description
	<p>Services that are not dependent on Agents instances</p> <ul style="list-style-type: none"> objects <p>Information about Agents instances or performance data collected from the monitoring targets</p> <ul style="list-style-type: none"> domain-types <p>Metadata of resources that are used for the API</p>
<i>object-type</i>	<p>The type of a resource or services.#</p> <p>Examples: RAID_PD of the Objects URI, AgentManager of the Service URL</p>
<i>action-identifier</i>	<p>The identifier of a specific action.#</p> <p>Example: refresh of the Service URI</p>
<i>query-string</i>	<p>Query string.</p> <p>Specifies refined conditions for object-type. Valid query strings are defined for each resource. The specifiable <i>query-string</i> vary for each resource. If a <i>query-string</i> other than an instance name or performance information with time granularity and how to obtain the performance information includes alphabetic characters, it is case-sensitive. The order in which a <i>query-string</i> is presented does not matter. An invalid <i>query-string</i> is ignored.</p> <ul style="list-style-type: none"> If the resource is performance data, the request is invalid. If the resource is other than performance data, the query string is ignored. <p>For details on how to specify query strings, see Obtaining performance data on page 3-18 in API functions on page 3-6. For details about query strings that can be specified when obtaining performance data, see Table 3-17 Specifiable keys and values in query strings for obtaining performance data on page 3-18 in API functions on page 3-6.</p>

#:

For details about resources that can be specified, see the URIs in [Table 3-1 List of the Tuning Manager API resources on page 3-2](#).

Example of specifying URI (when specifying the performance data for a RAID_PI_LDS record)

Note that line breaks are inserted in the example below to make it easy to read. However, do not enter line breaks when specifying the URI.

```
https://127.0.0.1:22016/TuningManager/v1/objects/RAID_PI_LDS?
hostName=HOST
&agentInstanceName=AgentInstance
&startTime=2013-12-01T13:00Z
&endTime=2013-12-01T15:00Z
&fields=RECORD_TIME%1FLDEV_NUMBER%1FRANDOM_TOTAL_IO_RATE
&LDEV_NUMBER=00:00:01
&LDEV_NUMBER=00:00:02
```

For details about resources that can be specified under <domain>, see [Chapter 3, List of resources on page 3-1](#).

Supported HTTP methods

For the operation methods that are defined by HTTP, Tuning Manager API supports the `GET` method and the `POST` method.

The methods that can be used vary depending on the resources to be handled. For details about which methods can be used, see [Chapter 3, List of resources on page 3-1](#).

Tuning Manager API version compatibility

The following table shows the compatibility between versions of Tuning Manager series products and this API (major and minor version numbers).

Table 2-2 Compatibility between versions of Tuning Manager series products and the API

Tuning Manager series product version	API version (majorVersion.minorVersion format)
Earlier than v8.0.0	Not supported
v8.0.0 or later	1.0
v8.1.3 or later	1.1

Request message

A request message sent from a web client to the web server consists of the information in the following table:

Table 2-3 Configuration of the request message

Item	Description
Request line	Specifies the information to be requested for the web server. For details, see Request line on page 2-4 .
Request header	Specifies the types of data that can be received by the web browser and the compression method. For details, see Request headers on page 2-6 .
Blank line	Indicates the end of a request header.
Request body	Specifies information to be sent to the web server by using the <code>POST</code> method. For the <code>GET</code> method, the body of the request is blank.

Request line

The following table shows the items in a request line:

Table 2-4 Configuration of the request line

(a) Method	(b) Space	(c) Request path (URI)	(d) Query	(e) Space	(f) HTTP version
---------------	--------------	---------------------------	--------------	--------------	---------------------

The following table describes the items making up the request line. Items (a) through (f) correspond with [Table 2-4 Configuration of the request line on page 2-5](#).

Table 2-5 Items that make up the request line

Item	Description	Example
(a) Method	The method used for the request	GET
(b) Space	A space character indicating a delimiter (single-byte only)	(A single byte space)
(c) Request path (URI)	A path indicating the resource	/TuningManager/v1/objects/RAID_PD
(d) Query	A query character string added to the request	?hostName=agent_1&agentInstanceName=VSPG1000_instance
(e) Space	A space character indicating a delimiter (single-byte only)	(A single byte space)
(f) HTTP version	A character string indicating the HTTP version	HTTP/1.1

Note that the maximum length of a request line is 8,190 bytes. If a request line that exceeds the maximum length is specified, a status code 414 (Request-URI Too Long) is returned.

Note

When a request is made, items (c) and (d) in the request path and the query [Table 2-4 Configuration of the request line on page 2-5](#) might be re-written by the internal processing and the request line might exceed the maximum length (8,190 bytes). In such a case, revise the request

path and the query values, and then perform the request by dividing it into multiple actions.

Request headers

The following table describes the headers that can be specified. Note that the names of header types are not case sensitive.

Table 2-6 Specifiable headers in Tuning Manager

Header	Description	Specification required
Authorization	This header is used for user authentication. For details on the authentication, see Security and authentication on page 2-8 .	Yes
Accept	This header is used to specify all media types of response data that can be received. If this header is omitted, the system assumes that an asterisk (*) is specified (that is, any media type of response data can be received). For the Tuning Manager API, if you explicitly specify media types by using the Accept header, you must specify <code>application/json</code> . In addition, you must specify <code>application/schema+json</code> to obtain metadata, and <code>text/csv</code> to obtain performance data.	If the Accept header is added, specify all media types required by the API resource. Note that you can omit the Accept header entirely.
Accept-Encoding	If the response is to be transmitted in compressed format, this header must be specified. For details on the transfer in compressed format, see Transmission in compressed format on page 2-8 .	No
Content-Type	Specify the media type for the request body.	You do not need to specify this header if you use the GET method. If you specify this header when using the GET method, this header is ignored. You must specify this header if you use the POST method.
Content-Length	Specify the size (in bytes) of the request body as a decimal number.	You do not need to specify this header if you use the GET method. However, if you specify this header, the size of the request body must be 0. If a non-zero value is specified, the

Header	Description	Specification required
		processing changes according to the web server specification. If you use the POST method, you must specify either the <code>Content-Length</code> header or the <code>Transfer-Encoding</code> header.
<code>Transfer-Encoding</code>	Specify the encoding method to be used when transferring the request body. In HTTP1.1, you can specify the chunked encoding method.	You do not need to specify this header if you use the GET method. However, if you specify this header, the size of the request body must be 0. If you use the POST method, you must specify either the <code>Content-Length</code> header or the <code>Transfer-Encoding</code> header.

An `Accept` header has the same interpretation whether it is specified on a single line or on multiple lines. The following are examples:

Example of specifying headers on multiple lines:

```
Accept: application/json
```

```
Accept: text/csv
```

```
Accept: application/xml
```

Example of specifying headers on a single line:

```
Accept: application/json, text/csv, application/xml
```

Note for headers that are not `Accept` headers: Headers that are specified on multiple lines or on single lines do not have the same interpretation. The following table describes the behavior when headers other than `Accept` headers are specified on multiple lines:

Table 2-7 Headers interpreted differently if specified on multiple lines

Header	Behavior when specified on multiple lines
<code>Authorization</code>	The first specified instance of the header has precedence.
<code>Accept-Encoding</code>	The line that specifies <code>gzip</code> has the first priority, and the line that specifies <code>identity</code> has the second priority.
<code>Content-Type</code>	The first specified instance of the header has precedence.
<code>Content-Length</code>	The behavior is determined by the behavior of the web server.
<code>Transfer-Encoding</code>	The behavior is determined by the behavior of the web server.
<code>Content-Length</code> and <code>Transfer-Encoding</code>	If both of these headers are specified, the <code>Transfer-Encoding</code> header takes priority.

Security and authentication

To use the API, the request message must include the `Authorization` header to send the authentication method and authentication information. The only authentication method that is supported is Basic authentication. Use the standard Basic authentication specification method to specify the header.

To use Basic authentication, include the `Authorization` header in the request message. In the `Authorization` header, specify `Basic` as the authentication method, followed by a *basic token*. A basic token is a Base64-encoded string that consists of a user name, a colon (`:`) (ASCII: `0x3A`), and a password. The following shows the format of the `Authorization` header for using Basic authentication.

The `Authorization` header in the request message for using Basic authentication:

```
Authorization: Basic basic-token
```

You can use either `http` or `https`, but we recommend accessing through `https` for improved security.

Transmission in compressed format

There are some cases in which access uses a low-throughput line (for example, in access from a remote location), or the data required by one request is too large. In such cases, the transmission time can be reduced by compressing the content. However, if a small amount of data is acquired over a high-throughput line, the compression process might cause a bottleneck. Therefore content is compressed only when requested from the API client.

Note that content compression can be used only when the version of the HTTP protocol is 1.1. The supported encoding methods are `gzip` and `identity`. Note that content is not compressed when encoded by using the `identity` encoding method.

The API client includes the `Accept-Encoding` header in a request message to report the encoding methods. You can specify multiple encoding methods in the API client and then assign a priority level to each method.

If an asterisk (`*`) is reported in the `Accept-Encoding` header, `gzip` is used as the encoding method.

For example, if you want either the `gzip` or `identity` method to be used as the compression method, you can specify the `Accept-Encoding` header in the request message as follows:

```
Accept-Encoding: gzip, identity
```

You can use content compression for all URIs, but the contents are only compressed if the size of the response body exceeds 100 bytes.

If content is compressed, the `Content-Encoding` header and the `Vary` header are included in the response message. The `Content-Encoding` header reports

the used encoding method to the client. Similarly, the `Vary` header reports that the response body was changed due to compression. However, if the encoding method is `identity`, the response message does not include the `Content-Encoding` header and the `Vary` header.

The following shows the `Content-Encoding` header and the `Vary` header included in the response message if content is compressed by using the `gzip` encoding method:

```
Content-Encoding: gzip
Vary: Accept-Encoding
```

Common objects

This section describes common objects that can be used independently of API resources. Note that the items in the field for all JSON objects (including common objects) are displayed in a random order.

Collection object

The `Collection` object is used to contain a set of objects to be returned. The following table describes the schema of the `Collection` object.

Table 2-8 Schema of the Collection object

Variable name	Type	Description
<code>items</code>	Object array	A set of objects

The following shows an example of the `Collection` object.

Example of the `Collection` object:

```
{ "items" : [{"type":"string", "value": "instance101"},
...
{"type":"string", "value": "instance200"}] }
```

Error object

If an error occurs in a request, both an `Error` object defined by the schema in the following table and a status code (see [Appendix A, Status Codes on page A-1.](#)) are returned. However, note that for API that return `Job` objects, the `Job` object might be returned depending on where the error occurred.

Table 2-9 Schema of the Error object

Variable name	Type	Description
<code>errorSource</code>	String	The URI sent by the API client (including the <code>GET</code> parameter)

Variable name	Type	Description
message	String	The error message (same message that is logged)
messageID	String	The message ID (same ID that is logged)
actions	String array	Actions that the API client must take

The following shows an example of the `Error` object.

Example of the `Error` object

```
{ "errorSource": "https://localhost:22016/TuningManager/v1/objects/AgentForRAID",
  "message": "The media type specified in the Content-Type header is unsupported.",
  "messageID": "KATR00111-E",
  "actions": ["Request does not contain the Content-Type header, or specified the media type is not supported by Tuning Manager Server."]}
```

Action object

The `Action` object is a container for the parameters for executing an action of a resource or service. The following table describes the schema of the `Action` object.

Table 2-10 Schema of the Action object

Variable name	Type	Description
name	String	The action name
url	String	The URI of the action to be executed
method	String	The HTTP method used to execute the action
type	String	The media type specified in the <code>Content-Type</code> header of the request to execute the action
parameters	Object array	The parameters needed for executing the action. The number of elements and the meaning of each element are defined for each action (API).

PrimitiveValue object

The `PrimitiveValue` object is a container that is used to treat a non-object variable as an object. The following table describes the schema of the `PrimitiveValue` object.

Table 2-11 Schema of the PrimitiveValue object

Variable name	Type	Description
type	String	The type of variable

Variable name	Type	Description
value	String	The value of the variable

The following shows an example of the `PrimitiveValue` object:

Example of the `PrimitiveValue` object:

```
{ "type" : "string",
  "value" : "Agent" }
```

Version object

This is a container for handling the version information. The following table describes the schema of the `Version` object.

Table 2-12 Schema of the Version object

Variable name	Type	Description
productName	string	Product name
productVersion	string	Product version
apiVersion	string	API version
description	string	Description of the API version

Job object

This is a container for handling the execution results of the Job object. The following table describes the schema of the `Job` object.

Table 2-13 Schema of the Job object

Variable name	Type	Description
status	String	The status indicating the result of processing. If the processing is completed normally, <code>Completed</code> is set. If an error occurs, <code>Error</code> is set.
errorID	String	The information that uniquely identifies the error that occurred. If an error occurs, the same string as the error ID output to the log is set. This variable is not added if processing ends normally.
errorDescription	String	An explanation of the error that occurred. If an error occurs, the same string that is output to the log is set. This variable is not added if processing ends normally.

List of resources

This chapter describes the details of the Tuning Manager API. Also, this chapter gives examples of the general procedure for obtaining performance data by using the API.

- [List of Tuning Manager API resources](#)
- [General procedure for obtaining performance data](#)

List of Tuning Manager API resources

The following table lists the resources of the Tuning Manager API. Note that the resources are shown as URIs.

Table 3-1 List of the Tuning Manager API resources

Use of the API		Request		Reference	Media types of response data
		HTTP method	URI		
This request obtains the detailed version information of the API.		GET	/TuningManager/v1/configuration/Version	Obtaining the detailed version information of the API on page 3-6	Version object
This request obtains a list of Agents management actions.		GET	/TuningManager/v1/services/AgentManager/actions	Obtaining a list of Agents administrative actions on page 3-8	A Collection object that has an Action object (without parameter variables) as a child
This request obtains parameters when the list of Agents is refreshed manually.		GET	/TuningManager/v1/services/AgentManager/actions/refresh	Obtaining the Agents list parameter when the manual refresh process is executed on page 3-9	Action object
This request executes the process to manually refresh the list of Agents.		POST	/TuningManager/v1/services/AgentManager/actions/refresh/invoke	Manually refreshing the Agents list on page 3-11	Job object
This request obtains a list of Agent for RAID instances.		GET	/TuningManager/v1/objects/AgentForRAID	Obtaining a list of Agent for RAID instances on page 3-15	A Collection object that has an AgentForRAID objects as a child
This request obtains the performance data of logical device.	PD_ELC records	GET	/TuningManager/v1/objects/RAID_PD_ELC	Obtaining performance data on page 3-18	CSV#
	PD_LDC records	GET	/TuningManager/v1/objects/RAID_PD_LDC	Obtaining performance data on page 3-18	CSV#

Use of the API		Request		Reference	Media types of response data
		HTTP method	URI		
	PD_LSEC records	GET	/TuningManager/v1/objects/RAID_PD_LSEC	Obtaining performance data on page 3-18	CSV#
	PI_LDE, PI_LDE1, PI_LDE2 and PI_LDE3 records	GET	/TuningManager/v1/objects/RAID_PI_LDE	Obtaining performance data on page 3-18	CSV#
	PI_LDS, PI_LDS1, PI_LDS2 and PI_LDS3 records	GET	/TuningManager/v1/objects/RAID_PI_LDS	Obtaining performance data on page 3-18	CSV#
This request obtains the performance data of Port.	PD_PTC records	GET	/TuningManager/v1/objects/RAID_PD_PTC	Obtaining performance data on page 3-18	CSV#
	PI_PTS records	GET	/TuningManager/v1/objects/RAID_PI_PTS	Obtaining performance data on page 3-18	CSV#
This request obtains the performance data of Parity Group.	PD_RGC records	GET	/TuningManager/v1/objects/RAID_PD_RGC	Obtaining performance data on page 3-18	CSV#
	PI_RGS records	GET	/TuningManager/v1/objects/RAID_PI_RGS	Obtaining performance data on page 3-18	CSV#
This request obtains the performance data of Storage System.	PD records	GET	/TuningManager/v1/objects/RAID_PD	Obtaining performance data on page 3-18	CSV#
	PI records	GET	/TuningManager/v1/objects/RAID_PI	Obtaining performance data on page 3-18	CSV#
	PI_LDA records	GET	/TuningManager/v1/objects/RAID_PI_LDA	Obtaining performance data on page 3-18	CSV#

Use of the API		Request		Reference	Media types of response data
		HTTP method	URI		
This request obtains the performance data of CLPR.	PD_CLPC records	GET	/TuningManager/v1/objects/RAID_PD_CLPC	Obtaining performance data on page 3-18	CSV#
	PI_CLCS records	GET	/TuningManager/v1/objects/RAID_PI_CLCS	Obtaining performance data on page 3-18	CSV#
	PI_CLPS records	GET	/TuningManager/v1/objects/RAID_PI_CLPS	Obtaining performance data on page 3-18	CSV#
This request obtains the performance data of CLPR Per MP Blade.	PI_CLMS records	GET	/TuningManager/v1/objects/RAID_PI_CLMS	Obtaining performance data on page 3-18	CSV#
This request obtains the performance data of Processor.	PD_UMS records	GET	/TuningManager/v1/objects/RAID_PD_UMS	Obtaining performance data on page 3-18	CSV#
	PI_PRCs records	GET	/TuningManager/v1/objects/RAID_PI_PRCs	Obtaining performance data on page 3-18	CSV#
This request obtains the performance data of CHA.	PI_CHS records	GET	/TuningManager/v1/objects/RAID_PI_CHS	Obtaining performance data on page 3-18	CSV#
This request obtains the performance data of Journal Group.	PI_JNLS records	GET	/TuningManager/v1/objects/RAID_PI_JNLS	Obtaining performance data on page 3-18	CSV#
This request obtains the performance data of Disk Drive.	PI_PDOS records	GET	/TuningManager/v1/objects/RAID_PI_PDOS	Obtaining performance data on page 3-18	CSV#

Use of the API		Request		Reference	Media types of response data
		HTTP method	URI		
This request obtains the performance data of Pool.	PD_PLC records	GET	/TuningManager/v1/objects/RAID_PD_PLC	Obtaining performance data on page 3-18	CSV#
	PD_PLF records	GET	/TuningManager/v1/objects/RAID_PD_PLF	Obtaining performance data on page 3-18	CSV#
	PD_PLR records	GET	/TuningManager/v1/objects/RAID_PD_PLR	Obtaining performance data on page 3-18	CSV#
	PI_PLS records	GET	/TuningManager/v1/objects/RAID_PI_PLS	Obtaining performance data on page 3-18	CSV#
This request obtains the performance data of V-VOL.	PD_VVC records	GET	/TuningManager/v1/objects/RAID_PD_VVC	Obtaining performance data on page 3-18	CSV#
	PD_VVF records	GET	/TuningManager/v1/objects/RAID_PD_VVF	Obtaining performance data on page 3-18	CSV#
This request obtains the performance data of Tier (Pool).	PD_PLTC records	GET	/TuningManager/v1/objects/RAID_PD_PLTC	Obtaining performance data on page 3-18	CSV#
	PD_PLTR records	GET	/TuningManager/v1/objects/RAID_PD_PLTR	Obtaining performance data on page 3-18	CSV#
	PD_PLTS records	GET	/TuningManager/v1/objects/RAID_PD_PLTS	Obtaining performance data on page 3-18	CSV#
	PI_PLTI records	GET	/TuningManager/v1/objects/RAID_PI_PLTI	Obtaining performance data on page 3-18	CSV#
This request obtains the performance data of	PD_VVTC records	GET	/TuningManager/v1/objects/RAID_PD_VVTC	Obtaining performance data on page 3-18	CSV#

Use of the API		Request		Reference	Media types of response data
		HTTP method	URI		
Tier (V-VOL).	PI_VVTI records	GET	/TuningManager/v1/objects/RAID_PI_VVTI	Obtaining performance data on page 3-18	CSV#
This request obtains metadata of the Collection object.		GET	/TuningManager/v1/domain-types/Collection	Obtaining metadata on page 3-28	JSON object
This request obtains metadata of the Error object.		GET	/TuningManager/v1/domain-types/Error	Obtaining metadata on page 3-28	JSON object
This request obtains metadata of the Version object.		GET	/TuningManager/v1/domain-types/Version	Obtaining metadata on page 3-28	JSON object
This request obtains metadata of the Action object.		GET	/TuningManager/v1/domain-types/Action	Obtaining metadata on page 3-28	JSON object
This request obtains metadata of the Job object.		GET	/TuningManager/v1/domain-types/Job	Obtaining metadata on page 3-28	JSON object
This request obtains metadata of the AgentForRAID object.		GET	/TuningManager/v1/domain-types/AgentForRAID	Obtaining metadata on page 3-28	JSON object

#:

Outputs the record information corresponding to the type of object specified in <object-type>.

API functions

This subsection describes the Tuning Manager API functionality. The API can be used to get performance data of a managed target system.

API usage examples provided in this manual use the `curl` command (version 7.19.7, Linux), which is free software. Note that line breaks have been inserted in the example to make it easier to read.

Obtaining the detailed version information of the API

The following table lists the properties for obtaining the detailed version information of the API.

Table 3-2 Properties for obtaining the detailed version information of the API

Item	Description
Resource URI	<i>scheme://host:port/TuningManager/version/configuration/Version</i>
HTTP method that can be used	GET
Required role	Any authenticated role
Response data media type	application/json

The following table describes the schema of response data.

Table 3-3 Schema of the Version object

Variable name	Type	Description
productName	String	The product name of Tuning Manager Server
productVersion	String	The product version of Tuning Manager - Server
apiVersion	String	The API version, including the minor version number, in the format <i>majorVersion.minorVersion</i> .
description	String	A brief explanation of the API version

The following shows the response body received when the detailed version information of the API is obtained.

Response body:

```
{
  "productName": "product-name",
  "productVersion": "product-version",
  "apiVersion": "api-version",
  "description": "explanation-of-the-API-version"
}
```

API usage example (obtaining the detailed version information of the API)

The following is an example of Tuning Manager series product version v8.0.0.

Request by using the `curl` command:

```
curl -v -H "Content-Type: application/json" -u system:manager -X
GET https://server_1:22016/TuningManager/v1/configuration/Version
```

Execution result for the request:

```
GET /TuningManager/v1/configuration/Version HTTP/1.1
Authorization: Basic c3lzdGVtOm1hbmFnZXI=
User-Agent: curl/7.19.7 (i386-redhat-linux-gnu) libcurl/7.19.7
NSS/3.12.9.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22016
Accept: */*
```

Content-Type: application/json

HTTP/1.1 200 OK

Date: Thu, 19 Jun 2014 04:51:52 GMT

Server: Cosminexus HTTP Server

Last-Modified: Thu, 29 May 2014 16:22:18 GMT

Cache-Control: private, max-age=86400

Transfer-Encoding: chunked

Content-Type: application/json; charset=utf-8

```
{
  "productName": "Hitachi Tuning Manager",
  "productVersion": "8.0.0",
  "apiVersion": "1.0",
  "description": "The initial version for supporting Tuning Manager
API"
}
```

Obtaining a list of Agents administrative actions

This subsection describes how to obtain a list of Agents administrative actions.

For all the actions that are provided by Agents administrative, this API stores the Action objects for the objects with omitted parameter fields to the Collection object, and then returns the object to API Client. You can use this Collection object (list of actions) to find out what kind of actions are available.

The following table lists the properties for obtaining a list of Agents administrative actions.

Table 3-4 Properties for obtaining a list of Agents administrative actions

Item	Description
Resource URI	<i>scheme://host:port/TuningManager/version/services/AgentManager/actions</i>
HTTP method that can be used	GET
Required role	Any authenticated role
Response data media type	application/json

The following table describes the format of the data included in the response body.

Table 3-5 Format of the response body received when a list of Agents administrative actions is obtained

Object name	Number of instances	Description
Collection	1	The Collection object that is the parent object of Action objects
- Action	0 to <i>n</i>	Action objects (without the parameters variables)

The following shows the response body received when a list of Agents administrative actions is obtained.

Response body:

```
{
  "items": [{"name": "action name",
             "url": "URI-of-the-action",
             "method": "HTTP-method-used-to-execute-the-action",
             "type": "media-type-specified-in-the-Content-Type-header-of-
the-request-to-execute-the-action"}
]
```

API usage example (obtaining Agents administrative actions)

Request by using the `curl` command:

```
curl -v -H "Content-Type: application/json" -u system:manager -X
GET https://server_1:22016/TuningManager/v1/services/AgentManager/
actions
```

Execution result for the request:

```
GET /TuningManager/v1/services/AgentManager/actions HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (i386-redhat-linux-gnu) libcurl/7.19.7
NSS/3.12.9.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22016
Accept: */*
Content-Type: application/json
```

```
HTTP/1.1 200 OK
Date: Thu, 19 Jun 2014 04:52:26 GMT
Server: Cosminexus HTTP Server
Cache-Control: no-cache
Transfer-Encoding: chunked
Content-Type: application/json;charset=utf-8
```

```
{
  "items": [{"name": "RefreshAgentInstancesParameters",
             "url": "https://server_1:22016/TuningManager/v1/services/
AgentManager/actions/refresh/invoke",
             "method": "POST",
             "type": "application/json"}
]
```

Obtaining the Agents list parameter when the manual refresh process is executed

This subsection describes how to obtain the parameter that is necessary for immediately updating the Agents list.

For a action provided by Tuning Manager API, this API returns (to API Client) the Action object whose default value has been specified in the parameters field. This Action object can be used, without change, as an argument of the API described in [Manually refreshing the Agents list on page 3-11](#).

The following table lists the properties for obtaining the Agents list parameters when the manual refresh process is executed.

Table 3-6 Properties for obtaining the Agents list parameters when the manual refresh process is executed

Item	Description
Resource URI	<i>scheme://host:port/TuningManager/version/services/AgentManager/actions/refresh</i>
HTTP method that can be used	GET
Required role	Any authenticated role
Response data media type	application/json

The following table describes the `Action` objects included in response data. These `Action` objects are used as a request body when an immediate update of the Agents list in [Manually refreshing the Agents list on page 3-11](#) below is executed.

Table 3-7 Action object used to obtain the Agents list parameters when the manual refresh process is executed

Variable name	Value
name	RefreshAgentInstancesParameters
url	<i>scheme://host:port/TuningManager/version/services/AgentManager/actions/refresh/invoke</i>
method	POST
type	application/json
parameters	PrimitiveValue object Number of arguments: 1 The only argument of this variable specifies the type of Agent products that are the target of the update. <ul style="list-style-type: none"> • type <ul style="list-style-type: none"> ◦ string: Indicates the variable type. • value <ul style="list-style-type: none"> ◦ All: Specifies all corresponding agents.

The following shows the response body received when obtaining parameters for manually refreshing the Agents list.

Response body:

```
{
  "name": "RefreshAgentInstancesParameters",
  "url": "URI-of-the-manually-refreshing-the-Agents-list",
  "method": "POST",
  "type": "application/json",
```



```

    "parameters": [{"type": "string", "value": "All"}]
  }

```

API usage example (obtaining the Agents list parameter when the manual refresh process is executed)

Request by using the `curl` command:

```

curl -v -H "Content-Type: application/json" -u system:manager -X
GET https://server_1:22016/TuningManager/v1/services/AgentManager/
actions/refresh

```

Execution result for the request:

```

GET /TuningManager/v1/services/AgentManager/actions/refresh HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (i386-redhat-linux-gnu) libcurl/7.19.7
NSS/3.12.9.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22016
Accept: */*
Content-Type: application/json

```

```

HTTP/1.1 200 OK
Date: Thu, 19 Jun 2014 04:53:01 GMT
Server: Cosminexus HTTP Server
Cache-Control: no-cache
Transfer-Encoding: chunked
Content-Type: application/json;charset=utf-8

```

```

{
  "name": "RefreshAgentInstancesParameters",
  "url": "https://server_1:22016/TuningManager/v1/services/
AgentManager/actions/refresh/invoke",
  "method": "POST",
  "type": "application/json",
  "parameters": [{"type": "string", "value": "All"}]
}

```

Manually refreshing the Agents list

This subsection describes how to manually refresh the Agent list.

The following table lists the properties for manually refreshing the Agents list.

Table 3-8 Properties for manually refreshing the Agents list

Item	Description
Resource URI	<i>scheme://host:port/TuningManager/version/services/AgentManager/actions/refresh/invoke</i>
HTTP method that can be used	POST
Request data media type	application/json
Request body	Action object indicated in Table 3-9 Action objects used when manually refreshing the Agents list on page 3-12 .
Required role	Any authenticated role

Item	Description
Response data media type	application/json

Table 3-9 Action objects used when manually refreshing the Agents list

Variable name	Value
name	RefreshAgentInstancesParameters
url	<i>scheme</i> :// <i>host</i> : <i>port</i> /TuningManager/ <i>version</i> /services/AgentManager/actions/refresh/invoke
method	POST
type	application/json
parameters	PrimitiveValue object Number of arguments: 1 The only argument of this variable specifies the type of Agent products that are the target of the update. All arguments after the first argument, if any, are ignored. You can specify the following values: <ul style="list-style-type: none"> • type <ul style="list-style-type: none"> ◦ string: Indicates the variable type. • value <ul style="list-style-type: none"> ◦ All: Specifies all corresponding agents. ◦ AgentForRAID: Specifies Agent for RAID.

Example of specifying a parameter:

```
{
  "name"      : "RefreshAgentInstancesParameters",
  "url"       : "https://localhost:22016/TuningManager/v1/
                services/AgentManager/actions/refresh/
                invoke",
  "method"    : "POST",
  "type"      : "application/json",
  "parameters" : [ { "type" : "string",
                    "value" : "AgentForRAID" } ]
}
```

When the update processing completes normally, the response body includes the Job objects indicated in the following table. If an error occurs, the response body might include the Job object or the Error object, depending on the cause of the error. For details, see [Table 3-11 Error responses for a manual refresh of the Agents list on page 3-13](#).

Table 3-10 Schema of the Job object

Variable name	Type	Description
status	String	The status that indicates the result of processing. If the processing finishes normally, <code>Completed</code> is set. If an error occurs, <code>Error</code> is set.
errorID	String	The information that uniquely identifies the error that occurred. If an error occurs, the same string as the error ID output to the log is set. This variable is not added if the processing ends normally.
errorDescription	String	An explanation of the error that occurred. If an error occurs, the same string that is output to the log is set. This variable is not added if processing ends normally.

The following shows the response body when the processing of manually refreshing the Agents list ends normally.

Response body:

```
{"status": "Completed"}
```

The following table describes responses that can be received if an error occurs during a manual refresh of the Agents list.

Table 3-11 Error responses for a manual refresh of the Agents list

Error that occurred	HTTP response			
	Status Code	Body		
		Error object	Job object	
		messageID	Status	errorID
Collection Manager is not running.	200	n/a	"Error"	KATR11002-E
The agent type specified in the first argument of the Action object parameters is not supported.	400	KATR11032-E	n/a	n/a

Legend:

n/a : Not applicable

API usage example (when refreshing lists of all Agents)

Request by using the `curl` command:

```
curl -v -u system:manager -H "Content-Type: application/json" -H "Content-Length: *" -H "Transfer-Encoding: chunked" -X
```

```
POST https://server_1:22016/TuningManager/v1/services/AgentManager/
actions/refresh/invoke -d
'{ "name" : "RefreshAgentInstancesParameters",
"url" : "https://server_1:22016/TuningManager/v1/services/
AgentManager/actions/refresh/invoke",
"method" : "POST", "type" : "application/json",
"parameters" : [ { "type" : "string", "value" : "All" } ] }'
```

Execution result for the request:

```
POST /TuningManager/v1/services/AgentManager/actions/refresh/invoke
HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (i386-redhat-linux-gnu) libcurl/7.19.7
NSS/3.12.9.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22016
Accept: */*
Content-Type: application/json
Content-Length: *
Transfer-Encoding: chunked

HTTP/1.1 200 OK
Date: Fri, 20 Jun 2014 12:31:06 GMT
Server: Cosminexus HTTP Server
Cache-Control: no-cache
Transfer-Encoding: chunked
Content-Type: application/json; charset=utf-8
```

```
{"status": "Completed"}
```

API usage example (when refreshing the list of Agent for RAID)

Request by using the curl command:

```
curl -v -u system:manager -H "Content-Type: application/json" -H
"Content-Length: *" -H "Transfer-Encoding: chunked" -X
POST https://server_1:22016/TuningManager/v1/services/AgentManager/
actions/refresh/invoke -d
'{ "name" : "RefreshAgentInstancesParameters",
"url" : "https://server_1:22016/TuningManager/v1/services/
AgentManager/actions/refresh/invoke",
"method" : "POST", "type" : "application/json",
"parameters" : [ { "type" : "string", "value" : "AgentForRAID" } ] }'
```

Execution result for the request:

```
POST /TuningManager/v1/services/AgentManager/actions/refresh/invoke
HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (i386-redhat-linux-gnu) libcurl/7.19.7
NSS/3.12.9.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22016
Accept: */*
Content-Type: application/json
Content-Length: *
Transfer-Encoding: chunked

HTTP/1.1 200 OK
Date: Fri, 20 Jun 2014 12:32:14 GMT
Server: Cosminexus HTTP Server
Cache-Control: no-cache
```

Transfer-Encoding: chunked
 Content-Type: application/json; charset=utf-8

```
{"status": "Completed"}
```

Obtaining a list of Agent for RAID instances

This subsection describes how to obtain a list of Agent for RAID instances.

The following table lists the properties for obtaining the Agents information.

Table 3-12 Properties for obtaining the Agents list

Item	Description
Resource URI	<i>scheme://host:port/TuningManager/version/objects/AgentForRAID</i>
HTTP method that can be used	GET
Required role	Any authenticated role
Response data media type	application/json

The following table describes the format of the data included in the response body.

Table 3-13 Format of the response body received when the Agents list is obtained

Object name	Number of instances	Description
Collection	1	The <code>Collection</code> object that is the parent object of <code>AgentForRAID</code> objects
- <code>AgentForRAID</code>	0 to <i>n</i>	<code>AgentForRAID</code> objects

The following table describes the schema of the `AgentForRAID` object.

Table 3-14 Schema of the AgentForRAID object

Variable name	Type	Description
<code>instanceName</code>	String	An Agent instance name
<code>hostName</code>	String	The monitoring-host name of the host that runs Agent
<code>ipAddr</code>	String	The IP address of the host on which Agent is operating
<code>dataModelVersion</code>	Decimal	The Agent data model version
<code>lastUpdateTime</code>	String	The last time the Agent information was successfully updated. The value is set in the extended ISO8601 format (<code>YYYY-MM-DDThh:mm:ssZ</code>). <i>hh</i> is specified in 24-hour format. <i>Z</i> indicates the UTC time. If the processing has not succeeded even once, a null value is set.

Variable name	Type	Description
storageVendor	String	The vendor name of the monitoring-target storage system# If the monitoring-target storage system does not exist, a null value is set.
storageModel	String	The model name of the monitoring-target storage system# If the monitoring-target storage system does not exist, a null value is set.
storageSerialNumber	String	The serial number of the monitoring-target storage system# If the monitoring-target storage system does not exist, a null value is set.

#:

Virtual storage machines are not included in the monitoring-target storage system.

The following shows the response body received when a list of Agents instances is obtained.

Response body:

```
{
  "items": [{
    "instanceName": "instance-name",
    "hostName": "monitoring-host-name",
    "ipAddr": "IP-address",
    "dataModelVersion": data-model-version,
    "lastUpdateTime": "last-update-time",
    "storageVendor": "storage-vendor-name",
    "storageModel": "storage-model-name",
    "storageSerialNumber": "serial-number"
  },
  :
  :
  ]
}
```

The following table describes the response that can be received if an error occurs during acquisition of a list of Agent for RAID instances.

Table 3-15 Error response during acquisition of a list of Agent for RAID instances

Error that occurred	HTTP response	
	Status code	Error ID included in the body
An internal error occurred.	500	KATR00101-E

API usage example (obtaining a list of Agent for RAID instances)

Request by using the `curl` command:

```
curl -v -u system:manager -X GET https://server_1:22016/  
TuningManager/v1/objects/AgentForRAID
```

Execution result for the request:

```
GET /TuningManager/v1/objects/AgentForRAID HTTP/1.1  
Authorization: Basic c3lzdGVtOmlhbmFnZXI=  
User-Agent: curl/7.19.7 (i386-redhat-linux-gnu) libcurl/7.19.7  
NSS/3.12.9.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2  
Host: server_1:22016  
Accept: */*
```

```
HTTP/1.1 200 OK  
Date: Fri, 20 Jun 2014 12:32:53 GMT  
Server: Cosminexus HTTP Server  
Last-Modified: Fri, 20 Jun 2014 12:32:14 GMT  
Cache-Control: no-cache  
Transfer-Encoding: chunked  
Content-Type: application/json;charset=utf-8
```

```
{  
  "items": [{  
    "instanceName": "VSPG1000_instance",  
    "hostName": "agent_1",  
    "ipAddr": "IP-address-of-agent_1",  
    "dataModelVersion": 8.6,  
    "storageVendor": "HITACHI",  
    "storageModel": "VSP G1000",  
    "storageSerialNumber": "10051",  
    "lastUpdateTime": "2014-06-20T12:32:14Z"  
  },  
  {  
    "instanceName": "USPV_instance",  
    "hostName": "agent_2",  
    "ipAddr": "IP-address-of-agent_2",  
    "dataModelVersion": 8.6,  
    "storageVendor": "HITACHI",  
    "storageModel": "USP V",  
    "storageSerialNumber": "10007",  
    "lastUpdateTime": "2014-06-20T12:32:14Z"  
  },  
  :  
  :  
  {  
    "instanceName": "HUSVM_instance",  
    "hostName": "agent_2",  
    "ipAddr": "IP-address-of-agent_2",  
    "dataModelVersion": 8.6,  
    "storageVendor": "HITACHI",  
    "storageModel": "HUS VM",  
    "storageSerialNumber": "210944",  
    "lastUpdateTime": "2014-06-20T12:32:14Z"  
  }  
}]  
}
```

Obtaining performance data

This subsection describes how to obtain the performance data that Agents collected from the monitoring targets. Note that performance data is provided in CSV format.

The `object-type` under the objects domain is formatted as *agent-type-record-name*. For example, if the `object-type` is RAID_PD_RGC, you can obtain the PD_RGC record information from Agent for RAID.

Note that performance data is obtained from Hybrid Store[#]. The retention period for the performance data can be changed. For the default values of the retention period and for details on how to change the retention period, see the *Tuning Manager Agent Administration Guide*.

#

When the system is operated by using Store database as well as when use of the Tuning Manager API is enabled, obtain the performance data from the performance data file when a Store database is used.

The following table describes the properties for obtaining performance data.

Table 3-16 Properties for obtaining performance data

Item	Description
Resource URI	<i>scheme://host:port/TuningManager/version/objects/object-type?query-string</i>
HTTP method that can be used	GET
Required role	Any authenticated role
Response data media type	text/csv

For *query-string*, enter a key and its value in pairs in the format *key=value*. You can enter multiple query strings. To enter multiple query strings, use an ampersand (&) as a separator. The following table lists the keys and types of values that can be specified. Note that query strings must be percent-encoded.

Table 3-17 Specifiable keys and values in query strings for obtaining performance data

Key	Value		Required
Key name	Specifiable value	Remarks	
hostName	Host name	The name of the Agents host This value is equivalent to the <code>hostName</code> value in Table 3-14 Schema of the AgentForRAID object on page 3-15 .	Yes

Key	Value		Required
Key name	Specifiable value	Remarks	
agentInstanceName	Agent instance name	The value that uniquely identifies an Agent instance managed by Agents This value is equivalent to the <code>instanceName</code> value in Table 3-14 Schema of the AgentForRAID object on page 3-15 .	Yes
startTime ^{#1}	A value that follows the following convention (the extended ISO8601 format) can be specified (Example: <code>startTime=2013-11-20T19:54Z</code>): <i>value = year-month-dayThour:minutez^{#2}</i> <i>year = 4DIGIT (0000-9999)</i> <i>month = 2DIGIT (01-12)</i> <i>day = 2DIGIT (01-31)</i> <i>hour = 2DIGIT (00-23)</i> <i>minute = 2DIGIT (00-59)</i> Note: DIGIT represents a decimal number (0-9) in the ABNF convention.	The start time (in UTC) of the period for which performance data is to be obtained	No
endTime ^{#1}	Same as <code>startTime</code> (Example: <code>endTime=2013-11-21T19:54Z</code>)	The end time (in UTC) of the period for which performance data is to be obtained	No
fields	A value that follows the following convention can be specified. <i>value = 1*VCHAR / value %1Fvalue^{#2}</i> To specify multiple values ^{#3} , use %1F as a separator ^{#4} .	Record field names For the field names of a record, see the description of records in the <i>Tuning Manager Hardware Reports Reference</i> .	No
record-field-name	A value that follows the following convention can be specified. <i>value = *VCHAR^{#2}</i> To specify multiple values ^{#3} , use %1F as a separator ^{#4} .	Specify the name of a field as a key, and the value of the field as the value for the key.	No
granularity ^{#5, #6}	The specifiable values are as follows: <ul style="list-style-type: none"> When obtaining non-aggregated data: raw 	The time granularity of the performance information to be obtained	No

Key	Value		Required
Key name	Specifiable value	Remarks	
	<ul style="list-style-type: none"> • When obtaining data summarized hourly: hour • When obtaining data summarized daily: day • When obtaining data summarized weekly: week • When obtaining data summarized monthly: month • When obtaining data summarized annually: year <p>When no value is specified, raw is assumed.</p> <p>For a record of the PD record type, specify raw or specify no value because summarized data does not exist.</p> <p>Note that this value is not case-sensitive.</p>		
accessMode#5, #6	<p>The specifiable values are as follows:</p> <ul style="list-style-type: none"> • When obtaining performance data by using the Timeline method: t • When obtaining performance data by using the Snapshot method: s <p>When no value is specified, t is assumed.</p> <p>Note that this value is not case-sensitive.</p> <p>Supplementary note</p> <p>Even if you specify t, if the conditions for obtaining the performance data by using the Timeline method are not satisfied, the performance data will be obtained by using the Snapshot method.#7</p>	<p>Obtaining data by executing a query</p> <p>For agents that use Hybrid Store, there are two methods for obtaining data by using the Tuning Manager API. When data is collected, if you specify the best method in accessMode based on the resources specified to be obtained and the collection period, the data collection performance will increase.</p> <p>Timeline method</p> <p>Specify this mode when you want to limit the data to be obtained to specific resources from among the resources being monitored, and obtain performance data for a certain period of time. This method is effective when, for example, you want to pay attention to specific resources and draw graphs.</p>	No

Key	Value		Required
Key name	Specifiable value	Remarks	
		Snapshot method Specify this mode when you want to perform batch collections of performance data of the resources being monitored. This method is effective when, for example, you want to store all of the collected data in a data warehouse all at once.	

#1

When you specify `startTime` and `endTime`, note the following:

- Specify both `startTime` and `endTime`, or neither of them. You cannot specify only one of these keys.
If you specify neither `startTime` nor `endTime`, the performance data of the latest date (for one snapshot) is returned.
- For `startTime`, specify a value smaller than that for `endTime`.

#2

Augmented Backus–Naur Form (ABNF).

#3

If multiple values can be specified for a key, you can separate the values by `%1F`, and you can also specify the values by specifying the same key name on multiple lines. These different notations have the same meaning. The different notations can be mixed in one request.

The following shows examples of specifying multiple values.

- When separating the values by using `%1F`:
`fields=WRITE_IO_COUNT%1FWRITE_XFER_RATE`
`LDEV_NUMBER=00:00:01%1F00:00:02`
- When specifying the same key name:
`fields=WRITE_IO_COUNT&fields=WRITE_XFER_RATE`
`LDEV_NUMBER=00:00:01&LDEV_NUMBER=00:00:02`

#4

`%1F` is unit separator in ASCII code (non-printable).

#5

This value can be specified only when the system is collecting performance data of an agent that is operated by using Hybrid Store.

#6

The data that can be obtained might vary depending on the values specified for "granularity" and "accessMode", because the timeline and snapshot methods delete performance data at different times after the retention period expires.

#7

The performance data in Timeline method does not exist for the period that was recovered by performing restoration or conversion. Therefore, if you want to reference the performance data that was recovered during that period by specifying `t` in `accessMode`, or reference the data without specifying anything, the performance data is obtained by the Snapshot method. This makes the response time longer because the performance data that was collected during normal operation after data recovery is compared in the same condition.

The following shows an example of a request to obtain performance data.

Example of a request:

```
https://127.0.0.1:22016/TuningManager/v1/objects/RAID_PI_LDS?
hostName=HOST
&agentInstanceName=AgentInstance
&startTime=2013-12-01T13:00Z
&endTime=2013-12-01T15:00Z
&fields=RECORD_TIME%1FLDEV_NUMBER%1FRANDOM_TOTAL_IO_RATE
&LDEV_NUMBER=00:00:01
&LDEV_NUMBER=00:00:02
```

The response body contains performance data in CSV format. The following table describes the variables used in the response body.

Table 3-18 Response body variables

Variable name	Description
SelectedFieldName_m	The name of a specific field#1
FieldDelimiter	The field separator. A comma (0x09) is used as recommended by RFC4180.
RecordDelimiter	The data line separator. An extra delimiter is added to the end of the data to show that the data ends. A CR-LF code (0x0D 0x0A) is used as recommended by RFC4180.
DataType_m	The data type of a specific field#1
SelectedFieldValue_lm	The value of a specific field that is selected by the function that selects data to be obtained#1, #2

#1

`m` represents the selected field number.

#2

1 represents the line number of the selection results.

Response body:

```
FILE = NAME <RecordDelimiter> TYPE <RecordDelimiter> DATA_LINES
<RecordDelimiter> <RecordDelimiter> ;
NAME = <SelectedFieldName> | NAME <FieldDelimiter> NAME ;
TYPE = <DataType> | TYPE <FieldDelimiter> TYPE ;
DATA_LINES = DATA | DATA_LINES <RecordDelimiter> DATA_LINES ;
DATA = <SelectedFieldValue> | DATA <FieldDelimiter> DATA ;
```

The following describes responses that can be received if an error occurs during acquisition of performance data.

Table 3-19 Error responses for acquisition of performance data

Error	HTTP response			Error ID included in the body
	Status code	Header		
		Name	Value	
The specified instance is not recognized by the Tuning Manager server.	404	There is no header to be added.		KATR12005-E
The specified instance was deleted from the agent host.	404			KATR11010-E
Requests are concentrated on the host on which the specified agent instance is running.	503	Retry-After	60	KATR12001-W
The specified record type either does not exist or is not supported.	404	There is no header to be added.		KATR11012-E
The request is invalid.	400			One of the following: <ul style="list-style-type: none"> • KATR11008-E • KATR11014-E • KATR11016-E • KATR11018-E • KATR11020-E • KATR12006-E • KATR12009-E
The setting for using Tuning Manager API in Agents is not enabled.	500			KATR12004-E
The version of Agents does not correspond to the version of the API being executed.	500			KATR12003-E

Error	HTTP response			
	Status code	Header		Error ID included in the body
		Name	Value	
A communication error (connection rejected) occurred between the Tuning Manager server and Agents.	500			KATR12007-E
A communication error (connection disabled) occurred between the Tuning Manager server and Agents.	500			KATR12008-E
The Agents returned a status code other than 403, and a media type that is neither <code>text/csv</code> nor <code>application/json</code> .	500			KATR12048-E
An internal contradiction (Tuning Manager server) occurred.	500			One of the following: <ul style="list-style-type: none"> • KATR00101-E • KATR00103-E • KATR12002-E
An internal contradiction (Agents) occurred.	500			KATR11022-E
An internal error occurred during the response return.	Cancels the response return.			

API usage example (when obtaining the PD record information of Agent for RAID)

Request by using the `curl` command:

```
curl -v -u system:manager -X GET https://server_1:22016/TuningManager/v1/objects/RAID_PD?
hostName=agent_1%26agentInstanceName=VSPG1000_instance
```

Execution result for the request:

```
GET /TuningManager/v1/objects/RAID_PD?
hostName=agent_1%26agentInstanceName=VSPG1000_instance HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (i386-redhat-linux-gnu) libcurl/7.19.7
NSS/3.12.9.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22016
Accept: */*
```

```
HTTP/1.1 200 OK
Date: Fri, 20 Jun 2014 13:09:05 GMT
Server: Cosminexus HTTP Server
Last-Modified: Fri, 20 Jun 2014 13:00:07 GMT
Keep-Alive: timeout=3, max=100
```

Cache-Control: no-cache
Transfer-Encoding: chunked
Content-Type: text/csv;charset=utf-8
X-Pad: avoid browser bug

DKC_SERIAL_NUMBER,VENDOR_ID,DKC_NAME,CACHE_MEMORY_CAPACITY,CACHE_MEMORY_INSTALLED_SIZE,MONITORED_SLPR_NUMBER,COLLECTION_TIME,INTERVAL,INPUT_RECORD_TYPE,RECORD_TIME
string(32),string(64),string(64),ulong,ulong,string(8),time_t,ulong,string(8),time_t
"10051","HITACHI","VSP G1000",469504,0,"",2014-06-20
13:00:07,3600,"PD",2014-06-20 13:00:07

API usage example (when obtaining the PI_LDS record information of Agent for RAID)

Request by using the curl command:

```
curl -v -u system:manager -X GET https://server_1:22016/  
TuningManager/v1/objects/RAID_PI_LDS?  
hostName=agent_1%26agentInstanceName=VSPG1000_instance
```

Execution result for the request:

```
GET /TuningManager/v1/objects/RAID_PI_LDS?  
hostName=agent_1%26agentInstanceName=VSPG1000_instance HTTP/1.1  
Authorization: Basic c3lzdGVtOmlhbmFnZXI=  
User-Agent: curl/7.19.7 (i386-redhat-linux-gnu) libcurl/7.19.7  
NSS/3.12.9.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2  
Host: server_1:22016  
Accept: */*
```

```
HTTP/1.1 200 OK  
Date: Fri, 20 Jun 2014 13:20:15 GMT  
Server: Cosminexus HTTP Server  
Last-Modified: Fri, 20 Jun 2014 13:20:01 GMT  
Keep-Alive: timeout=3, max=100  
Cache-Control: no-cache  
Transfer-Encoding: chunked  
Content-Type: text/csv;charset=utf-8  
X-Pad: avoid browser bug
```

```
LDEV_NUMBER,READ_IO_COUNT,READ_IO_RATE,READ_HIT_IO_COUNT,READ_HIT_RATE,  
WRITE_IO_COUNT,WRITE_IO_RATE,WRITE_HIT_IO_COUNT,WRITE_HIT_RATE,READ  
_MBYTES,READ_XFER_RATE,WRITE_MBYTES,WRITE_XFER_RATE,READ_TOTAL_RESPON  
SE,READ_RESPONSE_RATE,WRITE_TOTAL_RESPONSE,WRITE_RESPONSE_RATE,TOTAL  
RESPONSE_RATE,RANDOM_TOTAL_IO_RATE,SEQUENTIAL_TOTAL_IO_RATE,RANDOM_TO  
TAL_XFER_RATE,SEQUENTIAL_TOTAL_XFER_RATE,VIRTUAL_SERIAL_NUMBER,VIRTUA  
L_DKC_NAME,VIRTUAL_LDEV_NUMBER,INTERVAL,INPUT_RECORD_TYPE,RECORD_TIME  
string(16),ulong,float,ulong,float,ulong,float,ulong,float,ulong,floa  
t,ulong,float,double,float,double,float,float,float,float,float  
,string(32),string(32),string(16),ulong,string(8),time_t  
"00:00:00",0,0.0000000E+000,0,0.0000000E+000,0,0.0000000E  
+000,0,0.0000000E+000,0,0.0000000E+000,0,0.0000000E  
+000,0.0000000000000000E+000,0.0000000E+000,0.0000000000000000E  
+000,0.0000000E+000,0.0000000E+000,0.0000000E+000,0.0000000E  
+000,0.0000000E+000,0.0000000E+000,"", "", "", 301, "LDS", 2014-06-20  
13:20:01  
"00:00:01",0,0.0000000E+000,0,0.0000000E+000,0,0.0000000E  
+000,0,0.0000000E+000,0,0.0000000E+000,0,0.0000000E
```

```

+000,0.0000000000000000E+000,0.0000000E+000,0.0000000000000000E
+000,0.0000000E+000,0.0000000E+000,0.0000000E+000,0.0000000E
+000,0.0000000E+000,0.0000000E+000,"","","","301","LDS",2014-06-20
13:20:01
"00:00:02",0,0.0000000E+000,0,0.0000000E+000,0,0.0000000E
+000,0,0.0000000E+000,0,0.0000000E+000,0,0.0000000E
+000,0.0000000000000000E+000,0.0000000E+000,0.0000000000000000E
+000,0.0000000E+000,0.0000000E+000,0.0000000E+000,0.0000000E
+000,0.0000000E+000,0.0000000E+000,"","","","301","LDS",2014-06-20
13:20:01
:
:
"00:AF:00",0,0.0000000E+000,0,0.0000000E+000,0,0.0000000E
+000,0,0.0000000E+000,0,0.0000000E+000,0,0.0000000E
+000,0.0000000000000000E+000,0.0000000E+000,0.0000000000000000E
+000,0.0000000E+000,0.0000000E+000,0.0000000E+000,0.0000000E
+000,0.0000000E+000,0.0000000E+000,"","","","301","LDS2",2014-06-20
13:20:01

```

API usage example (when obtaining the PD_PTC record information of Agent for RAID)

Request by using the curl command:

```

curl -v -u system:manager -X GET https://server_1:22016/
TuningManager/v1/objects/RAID_PD_PTC?
hostName=agent_1%26agentInstanceName=VSPG1000_instance

```

Execution result for the request:

```

GET /TuningManager/v1/objects/RAID_PD_PTC?
hostName=agent_1%26agentInstanceName=VSPG1000_instance HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (i386-redhat-linux-gnu) libcurl/7.19.7
NSS/3.12.9.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22016
Accept: */*

```

```

HTTP/1.1 200 OK
Date: Fri, 20 Jun 2014 12:36:36 GMT
Server: Cosminexus HTTP Server
Last-Modified: Fri, 20 Jun 2014 12:00:07 GMT
Keep-Alive: timeout=3, max=100
Cache-Control: no-cache
Transfer-Encoding: chunked
Content-Type: text/csv;charset=utf-8
X-Pad: avoid browser bug

```

```

PORT_NUMBER,PORT_NAME,CHA_NAME,PORT_WWN,PORT_TYPE,PORT_ROLE,PORT_SPEE
D,SLPR_NUMBER,COLLECTION_TIME,INTERVAL,INPUT_RECORD_TYPE,RECORD_TIME
string(8),string(64),string(16),string(32),string(8),string(16),strin
g(8),string(8),time_t,ulong,string(8),time_t
"0","CL1-A","CHA-1EÜ","50060E8007274300","FIBRE","Target","Auto","",
2014-06-20 12:00:07,3600,"PTC",2014-06-20 12:00:07
"1","CL1-B","CHA-1EÜ","50060E8007274301","FIBRE","Target","Auto","",
2014-06-20 12:00:07,3600,"PTC",2014-06-20 12:00:07
"2","CL1-
C","CHA-1EL","50060E8007274302","FIBRE","External","8Gbps","",
2014-06-20 12:00:07,3600,"PTC",2014-06-20 12:00:07
"3","CL1-D","CHA-1EL","50060E8007274303","FIBRE","Target","Auto","",

```



```

2014-06-20 12:00:07,3600,"PTC",2014-06-20 12:00:07
:
:
"116","CL8-E","CHA-2RU","","FICON","Target","","",2014-06-20
12:00:07,3600,"PTC",2014-06-20 12:00:07
"117","CL8-F","CHA-2RU","","FICON","Target","","",2014-06-20
12:00:07,3600,"PTC",2014-06-20 12:00:07
"118","CL8-
G","CHA-2RL","50060E8007274376","FCoE","Target","10Gbps","","
2014-06-20 12:00:07,3600,"PTC",2014-06-20 12:00:07
"119","CL8-
H","CHA-2RL","50060E8007274377","FCoE","Target","10Gbps","","
2014-06-20 12:00:07,3600,"PTC",2014-06-20 12:00:07

```

API usage example (when obtaining the PI_PRCS record information of Agent for RAID by specifying startTime and endTime)

Request by using the curl command:

```

curl -v -u system:manager -X GET https://server_1:22016/
TuningManager/v1/objects/RAID_PI_PRCS?
hostName=agent_1%26agentInstanceName=VSPG1000_instance
%26startTime=2014-06-19T01:00Z%26endTime=2014-06-19T02:30Z

```

Execution result for the request:

```

GET /TuningManager/v1/objects/RAID_PI_PRCS?
hostName=agent_1%26agentInstanceName=VSPG1000_instance
%26startTime=2014-06-19T01:00Z%26endTime=2014-06-19T02:30Z HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (i386-redhat-linux-gnu) libcurl/7.19.7
NSS/3.12.9.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22016
Accept: */*

```

```

HTTP/1.1 200 OK
Date: Fri, 20 Jun 2014 13:10:42 GMT
Server: Cosminexus HTTP Server
Last-Modified: Thu, 19 Jun 2014 02:29:01 GMT
Keep-Alive: timeout=3, max=100
Cache-Control: private, max-age=864000
Transfer-Encoding: chunked
Content-Type: text/csv;charset=utf-8

```

```

PROCESSOR_ID,ADAPTOR_ID,CONTROLLER,PROCESSOR_TYPE,PROCESSOR_BUSY_RATE
,MAX_PROCESSOR_BUSY_RATE,MAX_BUFFER_LENGTH,BUFFER_IO_COUNT,MAX_BUFFER
_IO_COUNT,BUFFER_IO_RATE,MAX_BUFFER_IO_RATE,INTERVAL,INPUT_RECORD_TYP
E,RECORD_TIME
string(16),string(16),string(8),string(8),float,float,float,float,flo
at,float,float,ulong,string(8),time_t
"00","MPB0","","MP",8.8735523E-001,8.8735523E-001,6.5535002E
+004,1.0000000E+000,1.0000000E
+000,1.5259022E-003,1.5259022E-003,54,"PRCS",2014-06-19 01:14:01
"01","MPB0","","MP",1.2086518E+000,1.2086518E+000,6.5535002E
+004,1.0000000E+000,1.0000000E
+000,1.5259022E-003,1.5259022E-003,54,"PRCS",2014-06-19 01:14:01
"02","MPB0","","MP",9.0272198E-001,9.0272198E-001,6.5535002E
+004,1.0000000E+000,1.0000000E
+000,1.5259022E-003,1.5259022E-003,54,"PRCS",2014-06-19 01:14:01
:

```

```

:
"2E", "MPB5", "", "MP", 3.0648675E+000, 3.0648675E+000, 6.5535002E
+004, 1.0000000E+000, 1.0000000E
+000, 1.5259022E-003, 1.5259022E-003, 61, "PRCS", 2014-06-19 02:29:01
"2F", "MPB5", "", "MP", 3.1383426E+000, 3.1383426E+000, 6.5535002E
+004, 1.0000000E+000, 1.0000000E
+000, 1.5259022E-003, 1.5259022E-003, 61, "PRCS", 2014-06-19 02:29:01
"_Total", "MPB5", "", "MPB", 3.0528414E+000, 3.0528414E+000, 6.5535002E
+004, 1.0000000E+000, 1.0000000E
+000, 1.5259022E-003, 1.5259022E-003, 61, "PRCS", 2014-06-19 02:29:01

```

Obtaining metadata

This subsection describes the API resource that is used to obtain the metadata on JSON objects that are used by data-acquisition API resources other than the resource for obtaining performance data. The representation of metadata follows the JSON Schema.

Note that the metadata for resources that obtain performance data is always provided together with CSV headers. You cannot obtain only metadata.

The following table lists the properties for obtaining metadata.

Table 3-20 Properties for obtaining metadata

Item	Description
Metadata URI	<i>scheme://host:port/TuningManager/version/ domain-types/object-type</i>
HTTP method that can be used	GET
Required role	Any authenticated role
Response data media type	application/schema+json

Schema of response data:

The schema follows the JSON Schema for the individual object.

The following shows the response body received when metadata on the AgentForRAID object is obtained.

Response body:

```

{
  "title" : "AgentForRAID",
  "type" : "object",
  "properties" :
  {
    "instanceName" : { "type" : "string" },
    "hostName" : { "type" : "string" },
    "ipAddr" : { "type" : "string" },
    "dataModelVersion" : { "type" : "decimal" },
    "lastUpdateTime" : { "type" : "string" },
    "storageVendor" : { "type" : "string" },
    "storageModel" : { "type" : "string" },
    "storageSerialNumber" : { "type" : "string" }
  }
}

```

```

    },
    "required" : [ "instanceName", "hostName" ] }

```

The following table describes responses that can be received if an error occurs during acquisition of the metadata.

Table 3-21 Error responses for acquisition of metadata

Error	HTTP response			
	Status Code	Header		Error ID
		Name	Value	
The specified metadata either does not exist or is not supported.	404	There is no header to be added.		KATR10045-E
An internal contradiction occurred.	500			KATR00101-E
An internal error occurred during the response return.	Cancels the response return.			

API usage example (obtaining metadata)

Request by using the `curl` command:

```
curl -v -u system:manager -X GET https://server_1:22016/TuningManager/v1/domain-types/AgentForRAID
```

Execution result for the request:

```

GET /TuningManager/v1/domain-types/AgentForRAID HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (i386-redhat-linux-gnu) libcurl/7.19.7 NSS/
3.12.9.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22016
Accept: */*

```

```

HTTP/1.1 200 OK
Date: Fri, 20 Jun 2014 12:37:25 GMT
Server: Cosminexus HTTP Server
Last-Modified: Thu, 29 May 2014 16:22:18 GMT
Cache-Control: private, max-age=86400
Transfer-Encoding: chunked
Content-Type: application/schema+json;charset=utf-8

```

```

{
  "title" : "AgentForRAID",
  "type" : "object",
  "properties" :
  {
    "instanceName" : { "type" : "string" },
    "hostName" : { "type" : "string" },
    "ipAddr" : { "type" : "string" },
    "dataModelVersion" : { "type" : "decimal" },
    "lastUpdateTime" : { "type" : "string" },
    "storageVendor" : { "type" : "string" },
    "storageModel" : { "type" : "string" },
    "storageSerialNumber" : { "type" : "string" }
  }
}

```

```

},
"required" : [ "instanceName", "hostName" ]}

```

General procedure for obtaining performance data

This section gives an example of using the API to obtain performance data after a change in the instance information or storage system monitored by Agent for RAID. The example shows the order in which APIs are used until the performance data is obtained using the APIs, and shows how to specify the parameters. If the instance information or storage system monitored by Agent for RAID is changed, before you obtain the performance data, you need to manually refresh the Agents list by using the API.

The following figure shows an overview of the general procedure:

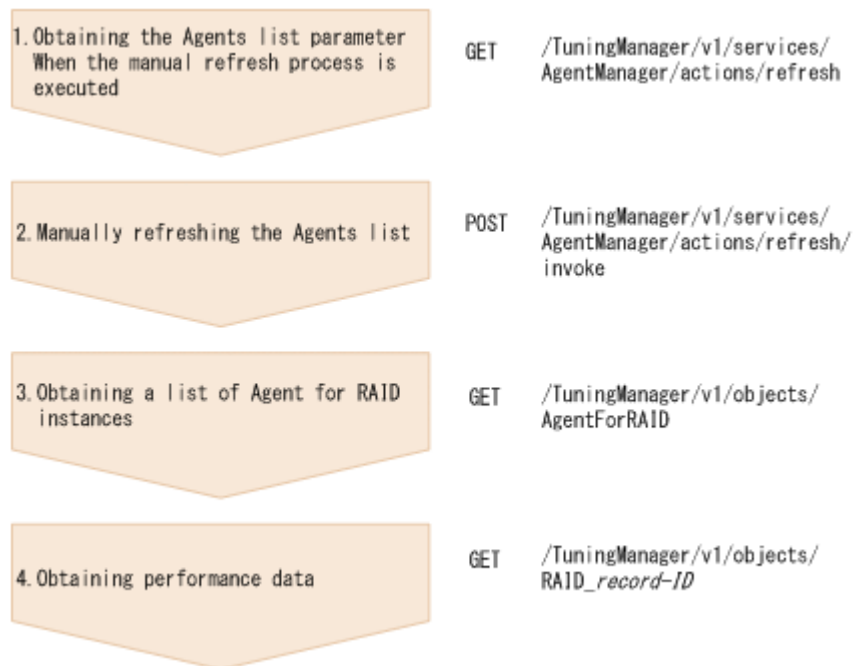


Figure 3-1 General procedure for obtaining performance data (after the monitoring-target storage system or instance information is changed)

Note that if the monitoring-target storage system or the instance information was not changed on Agent for RAID and you know the instance name of the monitoring target from which you want to obtain the performance data, you can obtain the performance data only in step 4 of [Figure 3-1 General procedure for obtaining performance data \(after the monitoring-target storage system or instance information is changed\)](#) on page 3-30 .

The following shows the general procedure for obtaining performance data of VSP G1000 by using the API, with an example of using the `curl` command, after changing the monitoring-target storage system or the instance information.

Example of using the curl command

1. Obtain the parameter information that is required to manually refresh the Agents list.

```
curl -v -H "Content-Type: application/json" -u system:manager -X
GET https://server_1:22016/TuningManager/v1/services/
AgentManager/actions/refresh
```

Response body:

```
{
  "name": "RefreshAgentInstancesParameters",
  "url": "https://server_1:22016/TuningManager/v1/services/
AgentManager/actions/refresh/invoke",
  "method": "POST",
  "type": "application/json",
  "parameters": [{"type": "string", "value": "All"}]
}
```

2. After the -d option, specify the response body obtained in step 1 as the request body to manually refresh the Agents list.

```
curl -v -u system:manager -H "Content-Type: application/json" -H
"Content-Length: *" -H "Transfer-Encoding: chunked" -X
POST https://server_1:22016/TuningManager/v1/services/
AgentManager/actions/refresh/invoke -d
'{"name" : "RefreshAgentInstancesParameters",
"url" : "https://server_1:22016/TuningManager/v1/services/
AgentManager/actions/refresh/invoke",
"method" : "POST", "type" : "application/json",
"parameters" : [ { "type" : "string", "value" : "All" } ] }'
```

Response body:

```
{"status": "Completed"}
```

3. Obtain the list of instance information for Agent for RAID.

```
curl -v -u system:manager -X GET https://server_1:22016/
TuningManager/v1/objects/AgentForRAID
```

Response body:

```
{
  "items": [{
    "instanceName": "VSPG1000_instance",
    "hostName": "agent_1",
    "ipAddr": "IP-address-of-agent_1",
    "dataModelVersion": 8.6,
    "storageVendor": "HITACHI",
    "storageModel": "VSP G1000",
    "storageSerialNumber": "10051",
    "lastUpdateTime": "2014-06-20T12:32:14Z"
  },
  {
    "instanceName": "USPV_instance",
    "hostName": "agent_2",
    "ipAddr": "IP-address-of-agent_2",
    "dataModelVersion": 8.6,
    "storageVendor": "HITACHI",
    "storageModel": "USP V",
    "storageSerialNumber": "10007",
    "lastUpdateTime": "2014-06-20T12:32:14Z"
  },
  :
  :
}
```

```

    {
      "instanceName": "HUSVM_instance",
      "hostName": "agent_2",
      "ipAddr": "IP-address-of-agent_2",
      "dataModelVersion": 8.6,
      "storageVendor": "HITACHI",
      "storageModel": "HUS VM",
      "storageSerialNumber": "210944",
      "lastUpdateTime": "2014-06-20T12:32:14Z"
    }
  ]
}

```

4. From the instance information obtained in step 3, specify the instance information from which you want to obtain the performance data as a query, and then obtain the performance data. In this example, the instance name `VSPG1000_instance` and the host name `agent_1` that are in the first object of the response body obtained by step 3, are specified as a query.

```

curl -v -u system:manager -X GET https://server_1:22016/
TuningManager/v1/objects/RAID_PD?
hostName=agent_1%26agentInstanceName=VSPG1000_instance

```

Response body:

```

LDEV_NUMBER, READ_IO_COUNT, READ_IO_RATE, READ_HIT_IO_COUNT, READ_HIT
_RATE, WRITE_IO_COUNT, WRITE_IO_RATE, WRITE_HIT_IO_COUNT, WRITE_HIT_R
ATE, READ_MBYTES, READ_XFER_RATE, WRITE_MBYTES, WRITE_XFER_RATE, READ
_TOTAL_RESPONSE, READ_RESPONSE_RATE, WRITE_TOTAL_RESPONSE, WRITE RESP
ONSE_RATE, TOTAL_RESPONSE_RATE, RANDOM_TOTAL_IO_RATE, SEQUENTIAL TOT
AL_IO_RATE, RANDOM_TOTAL_XFER_RATE, SEQUENTIAL_TOTAL_XFER_RATE, VIRT
UAL_SERIAL_NUMBER, VIRTUAL_DK_NAME, VIRTUAL_LDEV_NUMBER, INTERVAL, I
NPUT_RECORD_TYPE, RECORD_TIME
string(16), ulong, float, float, ulong, float, ulong, float, ulong,
float, float, float, double, float, double, float, float, float, float, flo
at, float, string(32), string(32), string(16), ulong, string(8), time_t
"00:00:00", 0, 0.0000000E+000, 0, 0.0000000E+000, 0, 0.0000000E
+000, 0, 0.0000000E+000, 0, 0.0000000E+000, 0, 0.0000000E
+000, 0.0000000000000000E+000, 0.0000000E+000, 0.0000000000000000E
+000, 0.0000000E+000, 0.0000000E+000, 0.0000000E+000, 0.0000000E
+000, 0.0000000E+000, 0.0000000E+000, "", "", "", 301, "LDS", 2014-06-20
13:20:01
"00:00:01", 0, 0.0000000E+000, 0, 0.0000000E+000, 0, 0.0000000E
+000, 0, 0.0000000E+000, 0, 0.0000000E+000, 0, 0.0000000E
+000, 0.0000000000000000E+000, 0.0000000E+000, 0.0000000000000000E
+000, 0.0000000E+000, 0.0000000E+000, 0.0000000E+000, 0.0000000E
+000, 0.0000000E+000, 0.0000000E+000, "", "", "", 301, "LDS", 2014-06-20
13:20:01
"00:00:02", 0, 0.0000000E+000, 0, 0.0000000E+000, 0, 0.0000000E
+000, 0, 0.0000000E+000, 0, 0.0000000E+000, 0, 0.0000000E
+000, 0.0000000000000000E+000, 0.0000000E+000, 0.0000000000000000E
+000, 0.0000000E+000, 0.0000000E+000, 0.0000000E+000, 0.0000000E
+000, 0.0000000E+000, 0.0000000E+000, "", "", "", 301, "LDS", 2014-06-20
13:20:01
:
:
"00:AF:00", 0, 0.0000000E+000, 0, 0.0000000E+000, 0, 0.0000000E
+000, 0, 0.0000000E+000, 0, 0.0000000E+000, 0, 0.0000000E
+000, 0.0000000000000000E+000, 0.0000000E+000, 0.0000000000000000E
+000, 0.0000000E+000, 0.0000000E+000, 0.0000000E+000, 0.0000000E

```

+000,0.0000000E+000,0.0000000E+000,"", "", "", 301, "LDS2",
2014-06-20 13:20:01

API Messages

This chapter explains the message format that are output during API execution, and the messages.

- [Message Format](#)
- [List of Output Destinations for Messages](#)
- [Messages](#)

Message Format

This section describes the format of messages that are output during API execution, and the format of the message explanations in this chapter.

Format of Output Messages

This subsection describes the format of messages that are output during API execution. Each message consists of a message ID followed by message text. The following is the message format:

KATRnnnnn-Y message-text

The message ID indicates the following:

- **KATR**: Indicates a message that is output when the Tuning Manager API is used.
- **nnnnn**: Serial number of the message. Main Console message numbers are from 00101 to 12999.
- **Y**: Type of message:
 - **E**: Error
Message issued when the system cancels processing
 - **W**: Warning
Message issued when the system resumes processing after message output
 - **I**: Information
Message in which the system provides the user with information

Message Description Format

This section describes the format of the message explanations in this chapter. The portion of a message text that is shown in italics represents information that is variable depending on the situation. The manual lists the messages in the order of the message IDs. The following illustrates the format of a message explanation:

message-ID message-text message-explanatory-text

(S) Explains the processing performed by the system.

(O) Explains the action the operator should take when the message is displayed.

List of Output Destinations for Messages

The table below lists the output destinations of messages that are output during API execution on the Tuning Manager server host and on the Agent host.

Table 4-1 List of output destinations of messages that are output during API execution (Tuning Manager server host)

Output Destinations	Message ID
<ul style="list-style-type: none"> Message log 	KATR10001, KATR10002, KATR10005, KATR10007 - KATR10010, KATR10012, KATR10015, KATR10017, KATR10018, KATR10020, KATR10041 - KATR10044, KATR11003 - KATR11007, KATR11028 - KATR11031
<ul style="list-style-type: none"> Error object# 	KATR00102, KATR10006, KATR10011, KATR10013, KATR10016
<ul style="list-style-type: none"> Error object# Message log 	KATR00101, KATR00103, KATR00105 - KATR00113, KATR10019, KATR10040, KATR10045, KATR11002, KATR11032, KATR12001 - KATR12009, KATR12048

#

The error object is returned to the API client.

Table 4-2 List of output destinations of messages that are output during API execution (Agent host)

Output Destinations	Message ID
<ul style="list-style-type: none"> Message log 	KATR10046 - KATR10051, KATR11026, KATR11027, KATR12010, KATR12014, KATR12025 - KATR12030, KATR12040, KATR12041, KATR12044, KATR12046, KATR12047, KATR12049 - KATR12052
<ul style="list-style-type: none"> Error object# Message log 	KATR00105, KATR11008 - KATR11025, KATR11033 - KATR11034

#

The error object is returned to the API client.

Messages

[Table 4-3 Messages that are output by API on page 4-3](#) explains the messages that are output by API from the Tuning Manager server host and the Agent host as well as the corrective actions to take.

Table 4-3 Messages that are output by API

Message ID	Message	Corrective Action
KATR00101-E	An unexpected error occurred. (<i>processing-name (license authentication agent information refreshing agent information reception performance data reception meta information reception)</i>)	An unexpected error occurred. (S) Stops processing, and returns an error response. (O) Contact the system administrator. If the problem cannot be resolved, detailed

Message ID	Message	Corrective Action
		investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.
KATR00102-E	An unexpected error occurred. (<i>processing-name (license authentication user authentication)</i>)	An unexpected error occurred. (S) Stops processing, and returns an error response. (O) Contact the Tuning Manager server administrator.
KATR00103-E	A database access error occurred. (<i>processing-name (agent information refreshing agent information reception performance data reception license authentication)</i>)	An error occurred while the database was being accessed. (S) Stops processing and then returns an error response. (O) Restart Tuning Manager Server. If the error occurs again, contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.
KATR00105-E	The specified HTTP method is not allowed. (<i>allowed methods =available-HTTP-method</i>)	The specified HTTP method is not allowed. (S) Stops processing. (O) Check the HTTP method.
KATR00106-E	An unexpected error occurred during initialization of Tuning Manager REST Application Server. Initialization failed. (<i>maintenance information = detailed-error-information</i>)	Initialization of Tuning Manager REST Application Server failed because of an unexpected error. (S) Stops initialization. However, services do not stop. (O) Restart Tuning Manager Server. If the error occurs again, contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask

Message ID	Message	Corrective Action
		you to collect maintenance information.
KATR00107-E	The request message does not include all the required headers. (missing header = <i>lacking-required-header</i>)	The request message does not include all required headers. (S) Stops processing, and returns an error response. (O) Check the request header.
KATR00108-E	No resource exists at the specified URL. (URL = <i>URL</i>)	The specified URL does not exist. (S) Stops processing, and returns an error response. (O) Check the URL.
KATR00109-E	The media type specified in the Accept header is unsupported.	This URL does not support the media type in the Accept header. (S) Stops processing, and returns an error response. (O) Check the Accept header.
KATR00110-E	The encoding format specified in the Accept-Encoding header is unsupported.	The encoding format specified in the Accept-Encoding header, is not supported in Tuning Manager server. (S) Stops processing, and returns an error response. (O) Check the Accept-Encoding header.
KATR00111-E	The media type specified in the Content-Type header is unsupported.	Request does not contain the Content-Type header, or specified the media type is not supported by Tuning Manager Server. (S) Stops processing, and returns an error response. (O) Check the Content-Type header.
KATR00112-E	The format of the JSON object specified in the request body is invalid.	The cause is one of the following: <ul style="list-style-type: none"> The specified JSON object in the request body is unsupported schema.

Message ID	Message	Corrective Action
		<ul style="list-style-type: none"> • The case which a Content-Length header is given, Tuning Manager Server could not read the full body due to the unmatched specified body size in the header and the actual body size. • The case which a Transfer-Encoding header is given, Tuning Manager Server could not read the full body due to the unmatched specified chunk-size in the body and the actual chunk size. <p>(S) Stops processing, and returns an error response.</p> <p>(O) Perform the following procedure:</p> <ul style="list-style-type: none"> - Check the request body. - Check the sizes in the Content-Length header and of the JSON object in the request body. - Check the specified chunk-size and the actual size of a chunk.
KATR00113-E	The request body is too long.	<p>The character string length of the request body exceeded the defined size.</p> <p>(S) Stops processing, and returns an error response.</p> <p>(O) Review the request body.</p>
KATR10001-E	An unexpected error occurred during license authentication. (details = <i>return-value-from-the-license-command</i>)	<p>License information could not be received.</p> <p>(S) Stops processing, and returns an error response.</p> <p>(O) Contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.</p>

Message ID	Message	Corrective Action
KATR10002-E	The system environment is invalid. (maintenance information = <i>command-name</i> , , <i>command-return-value</i>)	The system environment is invalid. (S) Stops processing, and returns an error response. (O) Contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.
KATR10005-E	Enter a license key or license key file.	A license key or license key file was not entered. (S) Stops processing, and returns an error response. (O) Register a valid license.
KATR10006-E	No valid license exists.	License information was not entered. (S) Stops processing, and returns an error response. (O) Contact the Tuning Manager server administrator.
KATR10007-E	The temporary license has expired. Enter a product license key or a product license key file.	The valid period of the temporary license key has expired. (S) Stops processing, and returns an error response. (O) Purchase a license, register it, and then retry the operation.
KATR10008-E	The emergency license has expired. Enter a product license key or a product license key file.	The valid period of the emergency license key has expired. (S) Stops processing, and returns an error response. (O) Purchase a license, register it, and then retry the operation.

Message ID	Message	Corrective Action
KATR10009-E	All of the subscription license has expired. Enter a product license key or a product license key file.	The valid period of all subscription license keys has expired. (S) Stops processing, and returns an error response. (O) Purchase a license, register it, and then retry the operation.
KATR10010-E	The user ID or password is incorrect.	The cause is one of the following: <ul style="list-style-type: none"> An invalid user ID or password was entered. The user does not have appropriate operation permissions. The user account is locked. (S) Stops processing, and returns an error response. (O) <ul style="list-style-type: none"> Check the entered user ID and password, and then enter them correctly. Perform the following procedure: Review the user's permissions as an administrator, and then grant the appropriate permission. Make sure the account is valid.
KATR10011-E	The user ID or password is incorrect.	The user ID or password is incorrect. (S) Stops processing, and returns an error response. (O) Check the entered user ID and password, and then enter them correctly.
KATR10012-E	Authentication failed.	The cause is one of the following: <ul style="list-style-type: none"> The SSO service is not running. Another unexpected problem occurred. (S)

Message ID	Message	Corrective Action
		<p>Stops processing, and returns an error response.</p> <p>(O)</p> <p>Wait a while, and then retry the operation. If the error occurs again, restart Tuning Manager Server. If this does not resolve the problem, contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.</p>
KATR10013-E	Authentication failed.	<p>Authentication failed.</p> <p>(S)</p> <p>Stops processing, and returns an error response.</p> <p>(O)</p> <p>Wait a while, and then retry the operation. If the error occurs again, contact the Tuning Manager server administrator.</p>
KATR10015-E	Communication with an external authentication server has failed.	<p>Communication with an external authentication server has failed.</p> <p>(S)</p> <p>Stops processing, and returns an error response.</p> <p>(O)</p> <p>Perform one of the following procedures:</p> <p>For LDAP:</p> <ol style="list-style-type: none"> 1. Check if you can access external authentication server by setting host, port and protocol in "exauth.properties". Also, check the network status and the configuration of external authentication server. 2. By using the hcmdsldapuser command, check that the information for the server names, specified in the "auth.server.name" attribute in "exauth.properties", is registered. 3. If StartTLS is used, check the SSL settings.

Message ID	Message	Corrective Action
		<p>4. If you cannot resolve the problem, contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.</p> <p>For RADIUS:</p> <ol style="list-style-type: none"> 1. Check if you can access external authentication server by setting host, port and protocol in "exauth.properties". Also, check the network status and the configuration of external authentication server. 2. By using the <code>hcmdsradiussecret</code> command, check that the information for the server names, specified in the "auth.server.name" attribute in "exauth.properties", is registered. 3. If you cannot resolve the problem, contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information. <p>For Kerberos:</p> <ol style="list-style-type: none"> 1. Make sure that the values specified for the <code>default_realm</code> attribute and the <code>kdc</code> attribute in <code>exauth.properties</code> are correct. 2. Make sure that the network is connected correctly. 3. Make sure that the external authentication server conforms to the Kerberos V5 protocol. 4. If you cannot resolve the problem, contact the system administrator. If the problem

Message ID	Message	Corrective Action
		<p>cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.</p> <p>For details on exauth.properties, see the Tuning Manager Installation Guide.</p>
KATR10016-E	A failure occurred during authentication.	<p>A failure occurred on the Tuning Manager server or authentication server.</p> <p>(S)</p> <p>Stops processing, and returns an error response.</p> <p>(O)</p> <p>Contact server administrator in Tuning Manager.</p>
KATR10017-E	Communication with an authentication server has failed.	<p>Common Component has not started or an error occurred during connection to Common Component.</p> <p>(S)</p> <p>Stops processing, and returns an error response.</p> <p>(O)</p> <p>Perform the following procedure:</p> <ol style="list-style-type: none"> 1. Check that Common Component is running. 2. Check that a connection error did not occur.
KATR10018-E	The Common Component has blocked Tuning Manager from accessing the database. Contact the server administrator.	<p>The RDAREA for the database is blocked.</p> <p>(S)</p> <p>Stops processing, and returns an error response.</p> <p>(O)</p> <p>Contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.</p>
KATR10019-E	The authentication parameter is invalid.	<p>An invalid character string is specified for the authentication method parameter.</p>

Message ID	Message	Corrective Action
		<p>(S) Stops processing, and returns an error response.</p> <p>(O) Confirm the parameter and retry authenticate.</p>
KATR10020-E	An unexpected error occurred during user authentication.	<p>The Common Component user information could not be obtained.</p> <p>(S) Stops processing, and returns an error response.</p> <p>(O) Contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.</p>
KATR10040-E	The Authentication HTTP header, which is required for authentication, is not specified.	<p>The Authorization header was not specified.</p> <p>(S) Returns an error response.</p> <p>(O) Specify Authorization header with valid value.</p>
KATR10041-I	Tuning Manager REST Application Server will now start.	Tuning Manager REST Application Server will now start.
KATR10042-I	Initialization of Tuning Manager REST Application Server finished.	Initialization of Tuning Manager REST Application Server finished.
KATR10043-I	Tuning Manager REST Application Server will now stop.	Tuning Manager REST Application Server will now stop.
KATR10044-I	Tuning Manager REST Application Server stopped.	Processing to stop Tuning Manager REST Application Server finished.
KATR10045-E	The specified meta information does not exist. (specified meta-information type = <i>specified-meta-information-type</i>)	<p>The specified meta-information does not exist.</p> <p>(O) Specify the correct meta-information and then execute again.</p>
KATR10046-E	An attempt to read an internal file has failed. (maintenance information = <i>properties-file-name, error-code</i>)	<p>Loading of an internal file failed. The cause of the error is indicated by one of the following error codes:</p>

Message ID	Message	Corrective Action
		<p>0: The file does not exist.</p> <p>1: There is no access permission for the file.</p> <p>2: The file format is incorrect.</p> <p>(S)</p> <p>Terminates the initial processing.</p> <p>(O)</p> <p>Back up the Agent, and then perform an overwrite installation of the Agent.</p>
KATR10047-E	An attempt to read a property file has failed. (property file name = <i>properties-file-name</i> , error code = <i>error-code</i>)	<p>Loading of the properties file failed. The cause of the error is indicated by one of the following error codes:</p> <p>0: The file does not exist.</p> <p>1: There is no access permission for the file.</p> <p>2: The file format is incorrect.</p> <p>(S)</p> <p>Terminates the initial processing.</p> <p>(O)</p> <p>Take the action appropriate for the error code:</p> <p>0: Locate the corresponding properties file to the appropriate location.</p> <p>1: Make sure that the file has the proper access permissions.</p> <p>2: Change to the correct format.</p> <p>For details about the properties file, see the <i>Tuning Manager Agent Administration Guide</i>.</p>
KATR10048-E	An unknown property has been specified. (property name = <i>property-name</i>)	<p>An unknown property has been set.</p> <p>(S)</p> <p>Continues checking other properties and then terminates the system.</p> <p>(O)</p> <p>Check and correct the property settings.</p>
KATR10049-E	The value specified for a property is invalid. (property file name = <i>properties-file-name</i> , property name = <i>property-name</i> , specified value = <i>specified-value</i>)	<p>The value of a property is incorrect.</p> <p>(S)</p> <p>Terminates the initial processing.</p> <p>(O)</p>

Message ID	Message	Corrective Action
		Correct the property setting.
KATR10050-E	An internal file is invalid. (maintenance information = <i>property-name</i>)	An unknown property has been set as a system property. (S) Continues processing. (O) Back up the Agent, and then perform an overwrite installation of the Agent.
KATR10051-E	An internal file is invalid. (maintenance information = <i>properties-file-name, property-name, specified-value</i>)	The value of a system property is incorrect. (S) Terminates the initial processing. (O) Back up the Agent, and then perform an overwrite installation of the Agent.
KATR11002-E	Collection Manager could not be accessed. Check the status of Collection Manager, and then try to refresh the agent information again.	Collection Manager has stopped. (S) Stops refreshing agent information. (O) Check the status of Collection Manager on the agent host.
KATR11003-W	Detection of Tuning Manager - Agent REST Web Service on an agent host failed. (host name = <i>host-to-be-detected</i> , protocol = <i>protocol-used-for-detection</i> , IP address = <i>IP-address-used-for-detection</i> , port = <i>port-used-for-detection</i> , status code = <i>HTTP-status-code-returned-from-the-agent-host</i>)	Possible causes are as follows: <ul style="list-style-type: none"> Because the version of the agent is old, Tuning Manager - Agent REST Web Service has not been installed. Tuning Manager - Agent REST Web Service on the agent host is not running. The settings of Tuning Manager - Agent REST Web Service on the agent host have been changed, but the necessary procedure has not been performed. Communication with Tuning Manager - Agent REST Web Service on the agent host is impossible. (S) Continues processing. (O)

Message ID	Message	Corrective Action
		<ul style="list-style-type: none"> • If the version of the agent is earlier than 8.0.0, upgrade the agent. • Check the status of Tuning Manager - Agent REST Web Service on the agent host. • If the settings of Tuning Manager - Agent REST Web Service on the agent host have been changed, make sure that you perform the necessary procedure on the Tuning Manager Server host. • Revise the network settings.
KATR11004-W	The monitoring target of an agent could not be identified. Make sure that the data management for the Tuning Manager API is enabled in the agent " <i>name-of-the-agent-instance-that-could-not-be-identified-as-a-monitoring-target</i> ".	<p>The function that outputs performance data files for the target agent might not be enabled.</p> <p>(S)</p> <p>Continues refreshing agent information.</p> <p>(O)</p> <p>Check the settings and status of the target agent.</p>
KATR11005-W	Tuning Manager - Agent REST Web Service on the agent could not be properly accessed. Check the status of Tuning Manager - Agent REST Web Service on the host " <i>host-name-of-the-agent-server-that-could-not-be-accessed</i> ". (status code = <i>HTTP-status-code-returned-when-access-failed</i>)	<p>Tuning Manager - Agent REST Web Service on the host returned an error response.</p> <p>(S)</p> <p>Continues refreshing agent information.</p> <p>(O)</p> <p>Check the status of Tuning Manager - Agent REST Web Service on the host.</p>
KATR11006-W	Tuning Manager - Agent REST Web Service on the agent could not be accessed. Check the status of Tuning Manager - Agent REST Web Service on the host " <i>host-name-of-the-agent-server-that-could-not-be-accessed</i> ".	<p>No response was returned from Tuning Manager - Agent REST Web Service on the host.</p> <p>(S)</p> <p>Continues refreshing agent information.</p> <p>(O)</p> <p>Check the status of Tuning Manager - Agent REST Web Service on the host.</p> <p>After you set HTTPS for Tuning Manager - Agent REST Web Service on the host, make sure that a certificate has been</p>

Message ID	Message	Corrective Action
		imported to the Tuning Manager Server.
KATR11007-I	Periodic refreshing of agent information will be skipped because the agent information is currently being refreshed.	A time to start periodic refreshing came while agent information was being refreshed.
KATR11008-E	An HTTP header is invalid. (header name = <i>HTTP-header-key</i> , value = <i>HTTP-header-value</i>)	An invalid HTTP header was specified. (O) After specify the HTTP header correctly, retry the request.
KATR11009-E	Specify the " <i>HTTP-header-key</i> " HTTP header correctly, and then retry the operation.	After specify the HTTP header correctly, retry the HTTP request.
KATR11010-E	The specified agent instance does not exist. (specified agent-instance name = <i>specified-instance-ID</i>)	The specified agent instance does not exist. (O) After specify the agent instance name correctly, retry the request.
KATR11011-E	Specify an agent instance that exists in Agent for RAID, and then retry the operation.	After specify the agent instance name correctly, retry the request.
KATR11012-E	The specified record ID does not exist. (record ID = <i>specified-record-ID</i>)	The specified record ID does not exist. (O) After specify the record ID correctly, retry the request.
KATR11013-E	Specify a record ID that exists in Agent for RAID, and then retry the operation.	After specify the record ID correctly, retry the request.
KATR11014-E	Required values in the query string are missing. (<i>key-names-that-must-be-specified-in-the-query-string</i>)	The query was short. (O) After specify the missing key, retry the HTTP request.
KATR11015-E	Specify the missing values, and then retry the operation.	After specify the missing key, retry the request.
KATR11016-E	Values that can not be specified in the query string are specified. (<i>values-that-were-specified-in-the-query-string</i>)	A value that should not specified was specified. (O) Check and correct the value, and then retry the request.
KATR11017-E	Remove the following items from the query string, and then retry the operation: <i>values-specified-in-the-query-string</i>	After removing the <i>values-that-were-specified-in-the-query-string</i> , retry the request.

Message ID	Message	Corrective Action
KATR11018-E	In the query string, a key is specified with an invalid value. (key = <i>key-name-specified-in-the-query-string</i> , value = <i>value-for-the-key-specified-in-the-query-string</i>)	The query was invalid. (O) Check and correct the value, and then retry the request.
KATR11019-E	Revise the value of the key, and then retry the operation. (key = <i>key-name-specified-in-the-query-string</i> , value = <i>value-for-the-key-specified-in-the-query-string</i>)	After checking the key and value (key: <i>key-name-specified-in-the-query-string</i> , value: <i>value-for-the-key-specified-in-the-query-string</i>), retry the HTTP request.
KATR11020-E	The specified time format is invalid. (key = <i>key-name-specified-in-the-query-string</i> , value = <i>value-for-the-key-specified-in-the-query-string</i>)	The specified time format is invalid. (O) Check the value. The time format is YYYY-MM-DDThh:mmZ.(e.g. 2013-12-03T21:52Z)
KATR11021-E	In the query string, check the value specified for the key. The time format to be used is "YYYY-MM-DDThh:mmZ" (for example, 2013-12-03T21:52Z). (key = <i>key-name-specified-in-the-query-string</i> , value = <i>value-for-the-key-specified-in-the-query-string</i>)	In the query string, check the value specified for the key. The time format to be used is "YYYY-MM-DDThh:mmZ" (for example, 2013-12-03T21:52Z). (key = <i>key-name-specified-in-the-query-string</i> , value = <i>value-for-the-key-specified-in-the-query-string</i>)
KATR11022-E	A server error occurred. (error detail = <i>error detail</i>)	A server error occurred. (O) Contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.
KATR11023-E	Contact the system administrator. If the problem cannot be resolved, contact Support Center, who might ask you to collect maintenance information.	Contact the system administrator. If the problem cannot be resolved, contact Support Center, who might ask you to collect maintenance information.
KATR11024-E	An attempt to load information failed, because the Agent instance is currently being initialized. (host name = <i>Host-name</i> , Agent type = <i>Agent-type</i> , instance name = <i>Agent-instance-name</i>)	An attempt to load information failed, because the Agent instance is currently being initialized. (O) Wait a while, and then try the operation again. If the error reoccurs, contact the system administrator. If the problem persists, to identify the cause and

Message ID	Message	Corrective Action
		resolve the problem, detailed investigation is required. Collect maintenance information, and then contact the customer support center.
KATR11025-E	Wait a while, and then try the operation again. If the error reoccurs, contact the system administrator. If the problem cannot be resolved, contact Support Center, who might ask you to collect maintenance information.	Information was collected from an agent instance that was being initialized.
KATR11026-E	An attempt to read data failed. (instance name = <i>specified-instance-ID</i> , record ID = <i>specified-record-ID</i> , data timestamp = <i>time</i>)	A performance data file open error has occurred. (S) After skip the data, continue to read. (O) Contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.
KATR11027-E	A data-file analysis error occurred. (instance name = <i>specified-instance-ID</i> , record ID = <i>specified-record-ID</i> , data timestamp = <i>time</i>)	A performance data file analysis error has occurred. (S) After skip the performance data file, continue to read. (O) If the problem occurs frequently, please contact an administrator or customer support.
KATR11028-I	Periodic refreshing of agent information will be disabled because the periodic-refresh interval for agent information is set to 0.	0 was set for the <code>rest.discovery.agent.interval</code> entry in the <code>user.properties</code> file. (S) Suppresses periodic refreshing of agent information.
KATR11029-W	An attempt to save agent information failed. (file name = <i>file-that-could-not-be-created-during-saving-of-agent-information</i>)	A file could not be created or renamed. (S) Continues processing. (O)

Message ID	Message	Corrective Action
		<p>For the directory in which the file indicated in the message is to be created, make sure that:</p> <ul style="list-style-type: none"> • The directory exists. • The directory is write-enabled.
KATR11030-W	An attempt to save agent information failed because deletion of old agent information failed. (file name = <i>file-that-could-not-be-deleted-during-saving-of-agent-information</i>)	<p>A file could not be deleted. (S) Continues processing. (O)</p> <p>For the directory in which the file indicated in the message is created, make sure that:</p> <ul style="list-style-type: none"> • The directory exists. • The directory is write-enabled.
KATR11031-W	An attempt to load agent information failed. (file name = <i>file-that-could-not-be-loaded-during-loading-of-agent-information</i>)	<p>A file could not be loaded. (S) Continues processing without agent information. (O)</p> <p>For the file indicated in the message, make sure the conditions below apply. After the problem is resolved, make sure that Tuning Manager - Agent REST Web Service on the agent instance and on each agent host has started, and then update the agent information.</p> <ul style="list-style-type: none"> • The file exists. • The file is read-enabled.
KATR11032-E	The specified agent type is unsupported. (agent type = <i>Agent type</i>)	<p>An unsupported agent type is specified. (S) Stops processing, and then returns an error response. (O) Confirm the agent types by Tuning Manager Server.</p>
KATR11033-E	Values that cannot be specified by using the current type of Performance database string are specified. (active mode = <i>Type-of-the-current-performance-database</i> , key that cannot be used = <i>Unsupported-keys-specified-in-the-query-character-string</i>)	<p>The specified request key is not supported by the type of the Performance database. (S) Stops processing and returns an error response. (O)</p>

Message ID	Message	Corrective Action
		Remove the following items from the query string, and then retry the operation. If the key is needed, change the type of Performance database to Hybrid Store.
KATR11034-E	Remove the following items from the query string, and then try the operation again. (<i>Unsupported-keys-specified-in-the-query-character-string</i>) If you want to use a key that you specify, change the type of the Performance database.	The specified request key is not supported by the type of the current Performance database. (S) Stops processing and returns an error response.
KATR12001-W	The agent host " <i>host-on-which-the-agent-having-the-specified-agent-ID-is-operating</i> " is experiencing a high volume of requests. Wait a while, and then retry the operation.	The host was overloaded with requests. (S) Stops sending requests to obtain performance data. (O) Make sure the host is not experiencing a large number of requests. If multiple APIs are being executed, reduce the number of APIs.
KATR12002-E	The agent information is invalid. Refresh the agent information, and then retry the operation. (host = <i>host-on-which-the-agent-having-the-specified-agent-ID-is-operating</i>)	An internal contradiction has occurred. (S) Stops sending requests to obtain performance data. (O) Refresh the agent information, and then try again.
KATR12003-E	The version of Tuning Manager - Agent REST Web Service on the target agent is invalid. (host = <i>host-on-which-the-agent-having-the-specified-agent-ID-is-operating</i>)	The combination of the API versions being used and Tuning Manager - Agent REST Web Service on the agent is not supported. (S) Stops sending requests to obtain performance data. (O) Change the API version. If you are using the latest API version, upgrade the version of the agent.
KATR12004-E	Tuning Manager - Agent REST Web Service was not detected on the target agent. (host = <i>host-on-which-the-agent-having-the-specified-agent-ID-is-operating</i>)	Possible causes are as follows: <ul style="list-style-type: none"> Because the version of the agent is old, Tuning Manager - Agent REST Web Service has not been installed.

Message ID	Message	Corrective Action
		<ul style="list-style-type: none"> • Tuning Manager - Agent REST Web Service on the agent host is not running. • The Tuning Manager server and the agent web application are not communicating. <p>(S) Stops sending requests to obtain performance data.</p> <p>(O) Perform the following procedure:</p> <ul style="list-style-type: none"> • If the version of the agent is earlier than 8.0.0, upgrade the version of the agent. • Check the status of Tuning Manager - Agent REST Web Service on the agent host. • Check the network settings. • If you changed the port number of Tuning Manager - Agent REST Web Service on the agent host, check and, if necessary, revise the Tuning Manager Server properties.
KATR12005-E	The specified agent does not exist. (agent type = <i>type-of-the-specified-agent</i> , instance name = <i>instance-name-of-the-specified-agent</i> , host name = <i>host-name-of-the-specified-agent</i>)	<p>The specified agent does not exist.</p> <p>(S) Stops sending requests to obtain performance data.</p> <p>(O) Check whether the agent information corresponding to the specified agent exists.</p>
KATR12006-E	A required parameter is not specified. (parameter name = <i>missing-parameter</i>)	<p>The GET parameter, which is necessary to obtain performance data, is not specified.</p> <p>(S) Stops sending requests.</p> <p>(O) Check whether the necessary parameter is specified.</p>
KATR12007-E	Connection from Tuning Manager to Tuning Manager - Agent REST Web Service on the agent host was refused. (host = <i>agent-host-that-could-not-be-accessed</i>)	<p>Access denial might be set for Tuning Manager - Agent REST Web Service on the agent host.</p> <p>(S) Stops sending requests.</p> <p>(O)</p>

Message ID	Message	Corrective Action
		Check and, if necessary, revise the settings for Tuning Manager - Agent REST Web Service on the agent host.
KATR12008-E	Tuning Manager - Agent REST Web Service on the agent host could not be accessed. (agent host = <i>agent host name</i>)	<p>Possible causes are as follows:</p> <ul style="list-style-type: none"> • Tuning Manager - Agent REST Web Service on the agent host is not running. • After the settings of Tuning Manager - Agent REST Web Service on the agent host were changed, agent information was not refreshed. • Communication with Tuning Manager - Agent REST Web Service on the agent host is impossible. <p>(S) Stops processing, and returns an error response.</p> <p>(O) Perform the following procedure:</p> <ul style="list-style-type: none"> • Check the status of Tuning Manager - Agent REST Web Service on the agent host. • If you changed the settings for Tuning Manager - Agent REST Web Service on the agent host, refresh the agent information. • Check and, if necessary, revise the network settings.
KATR12009-E	The value specified for the parameter is invalid. (parameter name = <i>Parameter-name</i> , value = <i>Specified-value</i>)	<p>The specified value is invalid.</p> <p>(S) return error response.</p> <p>(O) Specify a valid value.</p>
KATR12010-E	Initialization failed.	<p>An abnormality occurred during the initiation.</p> <p>(S) Stops processing.</p> <p>(O) Restart Tuning Manager - Agent REST Web Service. If the error occurs again, contact the system administrator. If the problem cannot be resolved, detailed investigation is required to</p>

Message ID	Message	Corrective Action
		determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.
KATR12014-E	Initialization is incomplete.	An attempt was made to continue the processing after the initiation failed. (S) Stops processing. (O) Restart Tuning Manager - Agent REST Web Service. If the error occurs again, contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information.
KATR12025-E	A property file cannot be read. (property file = <i>property-file</i>)	The property file could not be read. (S) Stops processing. (O) Check whether the property file exists, and verify the permissions needed to access the property file.
KATR12026-E	Check whether the property file " <i>property-file</i> " exists and its permissions.	Check whether the property file exists, and verify the permissions needed to access the property file.
KATR12027-E	A required property is not specified. (property file = <i>property-file</i> , key = <i>property-key</i>)	A necessary property is not specified. (S) Stops processing. (O) Check the property file settings.
KATR12028-E	Check the settings in the property file.	Check the property file settings.
KATR12029-W	A system property cannot be accessed. The default value will be used. (key = <i>property-key</i>)	The system property is not set. (S) Continues processing. (O) Set the system property.

Message ID	Message	Corrective Action
KATR12030-W	An environment variable cannot be accessed. The default value will be used. (name of environment variable = <i>environment-variable-name</i>)	The environment variable is not set. (S) Continues processing. (O) Set the environment variable.
KATR12040-E	An interrupt occurred.	An interruption occurred. (S) Stops processing.
KATR12041-E	Timeout occurred.	A timeout occurred. (S) Stops processing.
KATR12044-E	The number of threads exceeded the maximum.	There are a large number of requests, and the number of requests has exceeded the maximum number that can be executed. (S) Stops processing. (O) Wait a while, and then execute the requests.
KATR12046-E	An error occurred during termination processing.	An error occurred during end processing. (S) Continues processing. (O) Restart Tuning Manager - Agent REST Web Service. If the error occurs again, contact the system administrator. If the problem cannot be resolved, detailed investigation is required to determine the cause and resolve the problem. Contact the Support Center, who might ask you to collect maintenance information
KATR12047-W	The value of a property cannot be converted to a numerical value. The default value will be used. (key = <i>property-key</i>)	The value of the property that sets numerical values cannot be converted to a numerical value. (S) Continues processing. (O) Check and, if necessary, revise the value set for the property.

Message ID	Message	Corrective Action
KATR12048-E	Tuning Manager - Agent REST Web Service on the agent host replied with an unexpected response. (agent host = <i>agent host name</i>)	<p>The cause is one of the following:</p> <ul style="list-style-type: none"> • Tuning Manager - Agent REST Application Service is not running. • Another unexpected problem occurred. <p>(S) Stops processing, and returns an error response. (O) Check whether Tuning Manager - Agent REST Application Service is running on the agent host. If the problem cannot be resolved, contact Support Center, who might ask you to collect maintenance information.</p>
KATR12049-E	The format of content specified in the configuration file is invalid. (file path = <i>Path-of-the-settings-file</i> , row = <i>Row-in-the-settings-file-in-which-an-error-occurred</i> , specified content = <i>Character-string-in-the-settings-file-in-which-an-error-occurred</i>)	<p>The format of the setting specified in the setup file is invalid. (S) Stops processing. (O) Correct the setting in the relevant row.</p>
KATR12050-E	The configuration file contains a section name that is specified more than once. (file path = <i>Path-of-the-settings-file</i> , row = <i>Row-in-which-a-duplicate-section-name-is-found</i> , section name = <i>section-name</i>)	<p>The setup file contains a section name that is specified more than once. (S) Stops processing. (O) Change the section name.</p>
KATR12051-E	The configuration file contains a key name that is specified more than once. (file path = <i>Path-of-the-settings-file</i> , row = <i>Row-in-which-a-duplicate-key-name-is-found</i> , key name = <i>key-name</i>)	<p>The setup file contains a key name that is specified more than once. (S) Stops processing. (O) Change the key name.</p>
KATR12052-E	Failed to read a properties file. (file = <i>Name-of-the-properties-file</i> , section = <i>section-name</i> , label = <i>label-name</i>)	<p>Failed to read the properties file. (S) Stops processing. (O) Check whether you have permissions for the properties</p>

Message ID	Message	Corrective Action
		file. If you do, change the property settings.



A

Status Codes

This appendix describes the status codes of the response message output when the API is executed.

- [Status codes](#)

Status codes

The following table lists the status codes of the response message.

Table A-1 Status codes of the response message

Status code	Name	Description
200	OK	The request was processed normally.
400	Bad Request	The required header was omitted, the query character string is invalid, or the request body is invalid.
401	Unauthorized	The Authorization header was omitted, or authentication failed.
403	Forbidden	You do not have the required permissions.
404	Not Found	The requested resources do not exist.
405	Method Not Allowed	The requested method is not permitted by the resource.
406	Not Acceptable	Requests from the <code>Accept</code> header or from the <code>Accept-Encoding</code> header cannot be accepted.
414	Request-URI Too Long	The request line is too long.
415	Unsupported Media Type	An invalid media type was specified.
500	Internal Server Error	The required licenses are not registered.
		An error other than those listed above occurred in the Tuning Manager server or in Agents.
503	Service Unavailable	The service is temporarily unavailable due to the flow restriction of the Tuning Manager server and of Agents.

Specification differences depending on the Performance database being used

This appendix describes the specification differences of the Tuning Manager API based on whether you use Hybrid Store or a Store database as the Performance database.

- [Specification differences](#)

Specification differences

The specifications of the Tuning Manager API differ depending on whether you use Hybrid Store or a Store database as the Performance database. The following table lists the items that have specification differences, and the specifications for each Performance database for those items.

Table B-1 Specification differences depending on the Performance database being used

Item with a specification difference				Specifications when using a Store database	Specifications when using Hybrid Store
HTTP requests	Obtaining performance data	Query character strings	startTime	Searches at the time corresponding to the start time in the RECORD_TIME field of the record	Searches at the time corresponding to the start time in the DATETIME field of the record
			endTime	Searches at the time corresponding to the end time in the RECORD_TIME field of the record	Searches at the time corresponding to the end time in the DATETIME field of the record
			Filter conditions	Conditions are treated as enclosed characters when enclosed by double quotation marks (""). In addition, when specifying a value that contains spaces in a filter condition, you can URL encode the spaces to specify that condition.	Conditions are not valid, even when enclosed by double quotation marks ("") (double quotation marks are treated as characters). When specifying a value that contains spaces in a filter condition, specify the condition by URL encoding the spaces.
			granularity	Cannot be specified. When specified, the request becomes invalid.	Can be specified
			accessMode	Cannot be specified. When specified, the request becomes invalid.	Can be specified
HTTP responses	Obtaining performance data	Response body	Performance data	The summarized rule for PI_LDS, PI_LDS1, PI_LDS2, and TOTAL_RESPONSE_RATE field for the PI_LDS3 record is %.	The summarized rule for PI_LDS, PI_LDS1, PI_LDS2, and TOTAL_RESPONSE_RATE field for the PI_LDS3 record is HI_%.



Acronyms and Abbreviations

This manual uses the following acronyms and abbreviations:

A

ABNF

Augmented Backus–Naur form

API

Application Programming Interface

C

CSV

Comma Separated Value

G

GUI

Graphical User Interface

I

ID

IDentifier

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
---	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------

IP
Internet Protocol

J

JSON
JavaScript Object Notation

K

KDC
Key Distribution Center

L

LDAP
Lightweight Directory Access Protocol

O

OS
Operating System

R

RADIUS
Remote Authentication Dial In User Service

REST
Representational State Transfer

S

SSL
Secure Sockets Layer

SSO
Single Sign-On

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
---	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------

U

URI

Uniform Resource Identifier

URL

Uniform Resource Locator

UTC

Coordinated Universal Time

X

XML

eXtensible Markup Language

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
---	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
---	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------

Hitachi Data Systems

Corporate Headquarters

2845 Lafayette Street
Santa Clara, California 95050-2639
U.S.A.

www.hds.com

Regional Contact Information

Americas

+1 408 970 1000

info@hds.com

Europe, Middle East, and Africa

+44 (0)1753 618000

info.emea@hds.com

Asia Pacific

+852 3189 7900

hds.marketing.apac@hds.com



MK-92HC218-04