

Hitachi Command Suite

Tuning Manager

API リファレンスガイド

3021-9-042-E0

対象製品

Hitachi Tuning Manager 8.7.4

Hitachi Tuning Manager - Agent for RAID 8.7.4

Hitachi Tuning Manager - Agent for NAS 8.7.4

これらの製品には、他社からライセンスを受けて開発した部分が含まれています。

適用 OS の詳細については「ソフトウェア添付資料」でご確認ください。

輸出時の注意

本製品を輸出される場合には、外国為替及び外国貿易法の規制並びに米国輸出管理規則など外国の輸出関連法規をご確認の上、必要な手続きをお取りください。

なお、不明な場合は、弊社担当営業にお問い合わせください。

商標類

HITACHI, Cosminexus は、株式会社日立製作所の商標または登録商標です。

Excel は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

FICON は、世界の多くの国で登録された International Business Machines Corporation の商標です。

Linux は、Linus Torvalds 氏の日本およびその他の国における登録商標または商標です。

Microsoft は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Oracle と Java は、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。

Red Hat, and Red Hat Enterprise Linux are registered trademarks of Red Hat, Inc. in the United States and other countries.

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

RSA および BSAFE は、米国 EMC コーポレーションの米国およびその他の国における商標または登録商標です。

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

This product includes software developed by IAIK of Graz University of Technology.

Visual Basic は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Windows は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Windows Server は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

その他記載の会社名、製品名などは、それぞれの会社の商標もしくは登録商標です。

Hitachi Tuning Manager は、米国 EMC コーポレーションの RSA BSAFE(R)ソフトウェアを搭載しています。

This product includes software developed by Ben Laurie for use in the Apache-SSL HTTP server project.

Portions of this software were developed at the National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign.

This product includes software developed by the University of California, Berkeley and its contributors.

This software contains code derived from the RSA Data Security Inc. MD5 Message-Digest Algorithm, including various modifications by Spyglass Inc., Carnegie Mellon University, and Bell Communications Research, Inc (Bellcore).

Regular expression support is provided by the PCRE library package, which is open source software, written by Philip Hazel, and copyright by the University of Cambridge, England. The original software is available from <ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcre/>

This product includes software developed by Ralf S. Engelschall <rse@engelschall.com> for use in the mod_ssl project (<http://www.modssl.org/>).

This product includes software developed by Daisuke Okajima and Kohsuke Kawaguchi (<http://relaxngcc.sf.net/>).

This product includes software developed by the Java Apache Project for use in the Apache JServ servlet engine project (<http://java.apache.org/>).

This product includes software developed by Andy Clark.

Java is a registered trademark of Oracle and/or its affiliates.

HITACHI
Inspire the Next

 株式会社 日立製作所



マイクロソフト製品のスクリーンショットの使用について
マイクロソフトの許可を得て使用しています。

発行

2020年7月 3021-9-042-E0

著作権

All Rights Reserved. Copyright © 2014, 2020, Hitachi, Ltd.

目次

はじめに.....	13
対象読者.....	14
マニュアルの構成.....	14
マイクロソフト製品の表記について.....	14
読書手順.....	15
このマニュアルで使用している記号.....	15
ストレージシステムのサポート終了について.....	15
OS, 仮想化ソフトウェア, ブラウザーなどのサポートについて.....	16
1. Tuning Manager API の概要.....	17
1.1 Tuning Manager API の概要.....	18
1.2 http(s)経由で取得できる情報.....	19
1.3 http(s)経由で更新できる情報.....	20
1.4 API で監視対象のしきい値情報にアクセスできる条件.....	20
1.5 API でパフォーマンスデータにアクセスできる条件.....	20
1.6 Tuning Manager API の最大リクエスト数.....	21
1.7 Tuning Manager API を利用するために必要なインストールとセットアップの流れ.....	21
1.7.1 インストールとセットアップの流れ (非クラスタシステムで HTM - Agents を運用する場合)	22
1.7.2 インストールとセットアップの流れ (クラスタシステムで HTM - Agents を運用する場合)	23
2. Tuning Manager API の仕様.....	25
2.1 Tuning Manager API の共通仕様.....	26
2.2 サポートしている HTTP メソッド.....	28
2.3 リクエストメッセージ.....	28
2.3.1 リクエストライン.....	28
2.3.2 ヘッダー.....	29
2.4 セキュリティと認証.....	30
2.5 圧縮転送.....	31
2.6 Tuning Manager API のバージョン.....	31
2.7 共通オブジェクト.....	32
2.7.1 Collection オブジェクト.....	32
2.7.2 Error オブジェクト.....	32
2.7.3 Action オブジェクト.....	33
2.7.4 PrimitiveValue オブジェクト.....	33
2.8 ステータスコード.....	33

3. アラート機能の API.....	35
3.1 Tuning Manager API のリソース一覧.....	36
3.2 アラート定義情報の追加.....	36
3.3 アラート定義情報の削除.....	40
3.4 アラート定義情報の参照.....	43
3.5 アラート定義情報の同期実行時のパラメータの取得.....	48
3.6 アラート定義情報の同期.....	49
3.7 HTM - Agents 情報更新時のパラメーターの取得.....	53
3.8 HTM - Agents 情報の更新.....	54
3.9 HTM - Agents 情報取得時のパラメーターの取得.....	55
3.10 HTM - Agents 情報の取得.....	57
3.11 アラート定義情報設定時のオブジェクト.....	59
3.11.1 アラート定義情報設定時の注意事項.....	60
3.11.2 アラート定義型.....	60
3.11.3 target 型.....	60
(1) targetDef 型.....	61
(2) model 情報型.....	62
(3) raid 情報型.....	62
(4) vfp 情報型.....	62
(5) smu 情報型.....	63
(6) unified 情報型.....	63
(7) filter 型.....	64
(8) filterCondition 型.....	64
3.11.4 notice 型.....	64
(1) email 型.....	65
3.11.5 trigger 型.....	65
(1) monitoringTime 型.....	66
(2) condition 型.....	67
(3) frequency 型.....	67
3.11.6 definition 型.....	68
4. パフォーマンスデータを取得するための API.....	69
4.1 Tuning Manager API のリソース一覧.....	70
4.2 HTM - Agents 一覧の手動リフレッシュ処理実行時のパラメーターの取得.....	75
4.3 HTM - Agents 一覧の手動リフレッシュ処理の実行.....	76
4.4 HTM - Agents のインスタンス一覧の取得.....	79
4.5 HTM - Agent for RAID のインスタンス一覧の取得.....	85
4.6 パフォーマンスデータの取得.....	87
5. その他の Tuning Manager API.....	97
5.1 Tuning Manager API のリソース一覧.....	98
5.2 アラート定義情報の設定および運用のアクション一覧取得.....	98
5.3 HTM - Agents 管理のアクション一覧の取得（アラート機能の場合）.....	99
5.4 HTM - Agents 管理のアクション一覧の取得（パフォーマンスデータの取得の場合）.....	101
5.5 API の詳細バージョンの情報取得（アラート機能の場合）.....	102
5.6 API の詳細バージョンの情報取得（パフォーマンスデータの取得の場合）.....	103
5.7 メタ情報取得（パフォーマンスデータの取得の場合）.....	104

6. Tuning Manager API を実行するコマンド.....	107
6.1 コマンドの使用法.....	108
6.2 コマンド実行に必要な設定.....	109
6.2.1 Java 環境の設定.....	109
6.2.2 SSL の設定.....	109
6.3 コマンド実行の前提条件.....	110
6.4 htmrest.....	111
7. API のメッセージ.....	113
7.1 メッセージの形式.....	114
7.1.1 メッセージの出力形式.....	114
7.1.2 メッセージの記載形式.....	114
7.2 メッセージの出力先一覧.....	114
7.3 メッセージ一覧.....	115
付録 A Tuning Manager API の使用方法.....	143
A.1 アラート機能を設定・運用する流れ.....	144
A.2 アラート定義情報の作成手順.....	145
A.2.1 アラート定義情報を定義する前に確認すること.....	146
(1) アラート定義情報を適用する監視対象.....	146
(2) アラート定義情報を適用するリソースの絞り込み条件.....	146
(3) アラート定義のアクション情報.....	147
(4) アラート定義の監視条件情報.....	149
A.2.2 アラート定義情報を定義する際に使用できる文字について.....	151
(1) エスケープ文字.....	151
(2) 定義変数.....	151
(3) 特定の記号.....	151
A.2.3 アラート定義情報の定義例.....	152
A.3 LDEV 性能の Top5 のチャートを作成する手順.....	154
付録 B アラート定義情報のサンプル.....	161
B.1 監視対象がストレージシステムの場合.....	162
B.1.1 VSP Gx00 モデル, VSP Fx00 モデル, VSP E990, HUS VM, VSP 5000 シリーズ, VSP G1000, G1500, VSP F1500, および Virtual Storage Platform シリーズの場合.....	162
B.1.2 VSP Gx00 モデル, VSP Fx00 モデル, VSP E990, HUS VM, VSP 5000 シリーズ, VSP G1000, G1500, VSP F1500, および Virtual Storage Platform シリーズの場合 (TCP/IP 接続を使用して収集するとき).....	163
B.1.3 Universal Storage Platform V/VM シリーズ, Hitachi USP, および SANRISE H シリーズの場合.....	164
B.1.4 HUS100 シリーズ, および Hitachi AMS2000/AMS/WMS/SMS シリーズの場合.....	165
B.2 監視対象が NAS Platform の場合.....	166
付録 C Performance データベースによる仕様差異 (パフォーマンスデータを取得する場合).....	167
C.1 使用する Performance データベースによる仕様差異 (HTM - Agent for RAID の場合).....	168
付録 D このマニュアルの参考情報.....	169
D.1 関連マニュアル.....	170
D.2 このマニュアルでの表記.....	170
D.3 このマニュアルで使用している略語.....	171

目次

図 1-1 Tuning Manager API の概要図（アラート機能でのしきい値の監視の場合）	18
図 1-2 Tuning Manager API の概要図（パフォーマンスデータの取得の場合）	18
図 1-3 Tuning Manager API を利用するために必要なインストールとセットアップの流れ（非クラスタシステムで HTM - Agents を運用する場合）	22
図 1-4 Tuning Manager API を利用するために必要なインストールとセットアップの流れ（クラスタシステムで HTM - Agents を運用する場合）	23
図 6-1 コマンドの指定形式	108
図 A-1 アラート機能を設定・運用する流れ	144
図 A-2 evaluateAllData 変数に Y を指定した場合の email の例	148
図 A-3 evaluateAllData 変数に N を指定した場合の email の例	149
図 A-4 LDEV 性能の Top5 のチャートを作成するまでの流れ	155
図 A-5 LDEV 性能の Top5 のチャート（イメージ図）	156
図 A-6 LDEV 性能の Top5 の LDEV 番号の特定	159
図 A-7 LDEV 性能の Top5 のチャート	160

表目次

表 1-1 流れ図に示す作業項目の参照先一覧（非クラスタシステムで HTM - Agents を運用する場合）	22
表 1-2 流れ図に示す作業項目の参照先一覧（クラスタシステムで HTM - Agents を運用する場合）	23
表 2-1 URI の構成要素	26
表 2-2 リクエストメッセージの構成	28
表 2-3 リクエストラインの構成	28
表 2-4 リクエストラインを構成する項目	28
表 2-5 Tuning Manager シリーズで指定可能なヘッダー	29
表 2-6 複数行で指定した場合でも 1 行指定と同じと見なされないヘッダー	30
表 2-7 Tuning Manager シリーズの製品のバージョンと API のバージョンの対応（アラート機能で稼働監視する場合）	31
表 2-8 Tuning Manager シリーズの製品のバージョンと API のバージョンの対応（パフォーマンスデータを取得する場合）	32
表 2-9 Collection オブジェクトのスキーマ	32
表 2-10 Error オブジェクトのスキーマ	32
表 2-11 Action オブジェクトのスキーマ	33
表 2-12 PrimitiveValue オブジェクトのスキーマ	33
表 2-13 レスポンスメッセージのステータスコード	34
表 3-1 アラート機能のアラート定義情報を設定および運用する API	36
表 3-2 アラート定義情報追加時のスキーマ	37
表 3-3 アラート定義情報削除時のスキーマ	41
表 3-4 アラート定義取得時に指定可能なクエリ文字列のキーと値（アラート定義名の場合）	43
表 3-5 アラート定義取得時に指定可能なクエリ文字列のキーと値（監視対象の場合）	44
表 3-6 アラート定義同期時のリクエストボディのスキーマ	50
表 3-7 アラート定義同期時のレスポンスボディのスキーマ	50
表 3-8 情報の更新時のリクエストボディのスキーマ	54
表 3-9 情報の更新時のレスポンスボディのスキーマ	55
表 3-10 情報の取得時のリクエストボディのスキーマ	57
表 3-11 情報の取得時のレスポンスボディのスキーマ	57
表 3-12 アラート定義型のスキーマ	60
表 3-13 target 型のスキーマ	61
表 3-14 targetDef 型のスキーマ	61
表 3-15 model 情報型のスキーマ	62
表 3-16 raid 情報型のスキーマ	62
表 3-17 vfp 情報型のスキーマ	63
表 3-18 smu 情報型のスキーマ	63
表 3-19 unified 情報型のスキーマ	63
表 3-20 filter 型のスキーマ	64

表 3-21 filterCondition 型のスキーマ.....	64
表 3-22 notice 型のスキーマ.....	64
表 3-23 email 型のスキーマ.....	65
表 3-24 trigger 型のスキーマ.....	66
表 3-25 monitoringTime 型のスキーマ.....	66
表 3-26 condition 型のスキーマ.....	67
表 3-27 frequency 型のスキーマ.....	67
表 3-28 発生頻度と通知のタイミング.....	67
表 3-29 definition 型のスキーマ.....	68
表 4-1 Tuning Manager API のリソース一覧.....	70
表 4-2 HTM - Agents 一覧の手動リフレッシュ時の parameters.....	76
表 4-3 Job オブジェクトのスキーマ.....	77
表 4-4 HTM - Agents 一覧の手動リフレッシュ時のエラーレスポンス.....	77
表 4-5 HTM - Agents のインスタンス一覧取得時に指定可能なクエリ文字列のキーと値.....	79
表 4-6 HTM - Agents のインスタンス一覧取得時のデータフォーマット.....	81
表 4-7 Agents オブジェクトのスキーマ.....	81
表 4-8 nasNodes 属性のスキーマ.....	83
表 4-9 nasClusters 属性のスキーマ.....	84
表 4-10 HTM - Agents のインスタンス一覧取得時のエラーレスポンス.....	84
表 4-11 HTM - Agent for RAID のインスタンスの一覧取得時のデータフォーマット.....	85
表 4-12 AgentForRAID オブジェクトのスキーマ.....	85
表 4-13 HTM - Agent for RAID のインスタンス一覧取得時のエラーレスポンス.....	86
表 4-14 パフォーマンスデータ取得時に指定可能なクエリ文字列のキーと値.....	88
表 4-15 レスポンスボディの変数.....	91
表 4-16 パフォーマンスデータ取得時のエラーレスポンス.....	91
表 5-1 Tuning Manager API のリソース一覧.....	98
表 5-2 HTM - Agents 管理のアクション一覧取得時のデータフォーマット.....	101
表 5-3 Version オブジェクトのスキーマ（アラート機能の場合）.....	102
表 5-4 Version オブジェクトのスキーマ（パフォーマンスデータの取得の場合）.....	104
表 5-5 メタ情報取得時のエラーレスポンス.....	105
表 6-1 コマンドの文法の説明に使用する記号.....	108
表 6-2 htmrest コマンド引数のオプション.....	111
表 6-3 htmrest コマンドの戻り値.....	112
表 7-1 API 実行時に出力するメッセージの出力先一覧（Tuning Manager server ホスト）.....	114
表 7-2 API 実行時に出力するメッセージの出力先一覧（Agent ホスト）.....	115
表 7-3 API が出力するメッセージ.....	115
表 A-1 <条件式の判定記号> で指定できる値.....	150
表 A-2 エスケープ文字一覧.....	151
表 A-3 定義変数一覧.....	151
表 C-1 使用する Performance データベースによる仕様差異.....	168

はじめに

このマニュアルは、Tuning Manager シリーズで提供する API (Tuning Manager API) を利用するために必要な API リファレンスについて説明したものです。

API 利用時の構成などの概要については、マニュアル「Hitachi Command Suite Tuning Manager ユーザーズガイド」を、Tuning Manager server 側での設定方法については、マニュアル「Hitachi Command Suite Tuning Manager 運用管理ガイド」を、HTM - Agents 側での設定方法については、マニュアル「Hitachi Command Suite Tuning Manager - Agents」を参照してください。

- 対象読者
- マニュアルの構成
- マイクロソフト製品の表記について
- 読書手順
- このマニュアルで使用している記号
- ストレージシステムのサポート終了について
- OS, 仮想化ソフトウェア, ブラウザーなどのサポートについて

対象読者

- HTM - Agents の監視対象のしきい値を、API を利用して http(s)経由で監視したい方。
- HTM - Agents の監視対象の情報を、API を利用して http(s)経由で取得したい方。
- REST (Representational State Transfer) アーキテクチャについての基本的な知識をお持ちの方。

マニュアルの構成

このマニュアルは、次に示す章および付録から構成されています。なお、このマニュアルは、Windows および Linux の各 OS に共通のマニュアルです。

第 1 章 Tuning Manager API の概要

Tuning Manager API の概要について説明しています。

第 2 章 Tuning Manager API の仕様

Tuning Manager API の仕様について説明しています。

第 3 章 アラート機能の API

アラート機能のアラート定義情報を設定および運用する API について説明しています。

第 4 章 パフォーマンスデータを取得するための API

パフォーマンスデータを取得する前に実行が必要な API、およびパフォーマンスデータを取得する API について説明しています。

第 5 章 その他の Tuning Manager API

その他の API について説明しています。

第 6 章 Tuning Manager API を実行するコマンド

Tuning Manager API を実行するコマンドについて説明しています。

第 7 章 API のメッセージ

Tuning Manager API のメッセージについて説明しています。

付録 A Tuning Manager API の使用方法

Tuning Manager API の使用方法について説明しています。

付録 B アラート定義情報のサンプル

アラート定義情報のサンプルを記載しています。

付録 C 使用する Performance データベースによる仕様差異

使用する Performance データベースの違いによる Tuning Manager API の仕様差異について説明しています。

付録 D このマニュアルの参考情報

このマニュアルを読むに当たっての参考情報について説明しています。

マイクロソフト製品の表記について

このマニュアルでは、マイクロソフト製品の名称を次のように表記しています。

表記	製品名
Microsoft Office Excel	Microsoft(R) Office Excel
Windows	次の製品を区別する必要がない場合の表記です。 <ul style="list-style-type: none"> Windows Server 2008 Windows Server 2012
Windows Server 2008	Tuning Manager server がサポートしている Microsoft(R) Windows Server(R) 2008 の総称です。エディションは問いません。
Windows Server 2012	Tuning Manager server がサポートしている Microsoft(R) Windows Server(R) 2012 の総称です。エディションは問いません。

読書手順

このマニュアルは、利用目的に合わせて章を選択して読むことができます。利用目的別にお読みいただくことをお勧めします。

マニュアルを読む目的	記述箇所
Tuning Manager API を利用する上での概要について知りたい。	1 章
Tuning Manager API の仕様について知りたい。	2 章
各 Tuning Manager API の詳細について知りたい。	3 章, 4 章, 5 章
コマンドを使って Tuning Manager API を実行する方法について知りたい。	6 章
Tuning Manager API を使用して出力されるメッセージについて知りたい。	7 章

このマニュアルで使用している記号

このマニュアルで使用している記号を次に示します。

記号	意味
{ }	この記号で囲まれている複数の項目の中から、必ず 1 組の項目が該当することを示します。項目の区切りは で示します。 (例) {A B C} は「A, B, または C のどれかが該当する」ことを示す。
<斜体>	可変値であることを示します。
[]	任意指定 (省略可能) であることを示します。

ストレージシステムのサポート終了について

バージョン 8.5.3 から、次に示すストレージシステムのサポートを終了しました。サポートを終了したストレージシステムに関するマニュアル中の記載は無視してください。マニュアルでの表記については、「D.2 このマニュアルでの表記」を参照してください。

バージョン 8.6.1 からサポート終了

- Hitachi Universal Storage Platform 100
- Hitachi Universal Storage Platform 600
- Hitachi Universal Storage Platform 1100

- Hitachi Universal Storage Platform H10000
- Hitachi Universal Storage Platform H12000
- Hitachi network Storage Controller

バージョン 8.5.3 からサポート終了

- Hitachi Adaptable Modular Storage シリーズ
 - Hitachi Adaptable Modular Storage 1000
 - Hitachi Adaptable Modular Storage 500
 - Hitachi Adaptable Modular Storage 200
 - BladeSymphony 専用エントリークラスディスクアレイ装置 BR150
- Hitachi Workgroup Modular Storage シリーズ
 - Hitachi Workgroup Modular Storage シリーズ
 - BladeSymphony 専用エントリークラスディスクアレイ装置 BR50
- Hitachi Tape Modular Storage シリーズ

OS, 仮想化ソフトウェア, ブラウザーなどのサポートについて

OS, 仮想化ソフトウェア, ブラウザーなどの最新のサポート状況は, 「ソフトウェア添付資料」を参照してください。

サポートが終了したソフトウェアに関するマニュアル中の記載は無視してください。

新しいバージョンをサポートしたソフトウェアについては, 特に記載がないかぎり, 従来サポートしているバージョンと同等のものとしてサポートします。

Tuning Manager API の概要

この章では、Tuning Manager シリーズが提供している API (Tuning Manager API) について、概要を説明します。

- 1.1 Tuning Manager API の概要
- 1.2 http(s)経由で取得できる情報
- 1.3 http(s)経由で更新できる情報
- 1.4 API で監視対象のしきい値情報にアクセスできる条件
- 1.5 API でパフォーマンスデータにアクセスできる条件
- 1.6 Tuning Manager API の最大リクエスト数
- 1.7 Tuning Manager API を利用するために必要なインストールとセットアップの流れ

1.1 Tuning Manager API の概要

Tuning Manager シリーズでは、http(s)経由で API を利用して次のことができます。

- HTM - Agents が収集した監視対象の情報に対してしきい値を監視できます。
- HTM - Agents が収集した監視対象の情報を取得できます。

この API を Tuning Manager API と呼びます。

図 1-1 Tuning Manager API の概要図（アラート機能でのしきい値の監視の場合）

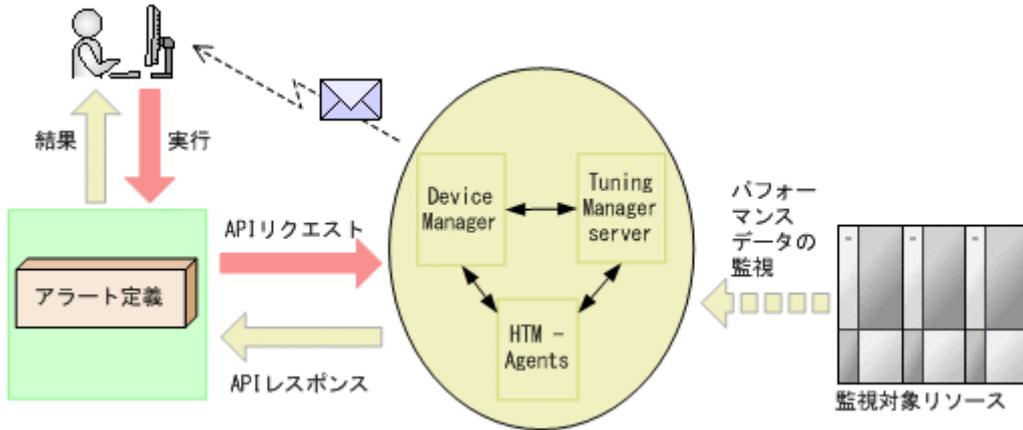
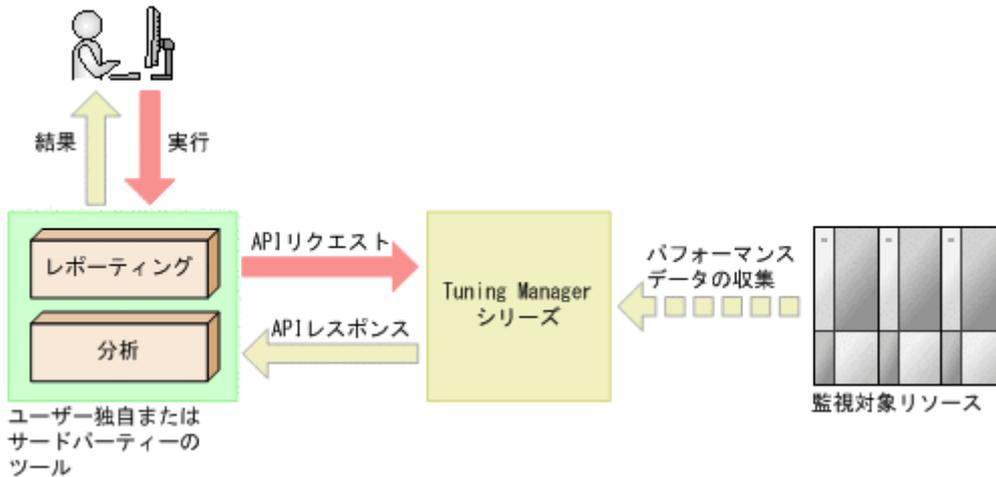


図 1-2 Tuning Manager API の概要図（パフォーマンスデータの取得の場合）



API を利用する利点を次に示します。

- ネットワークを経由してリモートから情報を取得できる。
- アラート機能で監視対象のしきい値を監視できる。
- 大量のパフォーマンスデータを高速で取得できる。
- 必要な情報だけを取得して独自にレポートニングできる。
- ユーザー独自のツールやサードパーティーのツールで、API で取得したパフォーマンスデータを分析できる。

なお、Tuning Manager API は、REST (Representational State Transfer) アーキテクチャスタイルに従います。REST アーキテクチャスタイルに従った API には次のような利点があります。

- プラットフォーム、および言語に依存しない。

- HTTP の標準インターフェースに従う。
- HTTP 向けのセキュリティ対策ツールを利用できる。
- 標準の HTTP のステータスコードが戻るのでエラー時に対処しやすい。
- リソースごとに URI (Uniform Resource Identifier) を持っているので、リソースにアクセスしやすい。またリソース間の関連を理解しやすい。

API を利用してアラート機能で監視対象の稼働状態を監視したり、取得したりできるパフォーマンスデータは、次に示す HTM - Agents が収集した情報です。

- HTM - Agent for RAID
- HTM - Agent for NAS

API を利用する場合のシステム構成および設定については、次のマニュアルを参照してください。

- システム構成 :
マニュアル「Hitachi Command Suite Tuning Manager ユーザーズガイド」
- Tuning Manager server で実施する設定 :
マニュアル「Hitachi Command Suite Tuning Manager 運用管理ガイド」
- エージェントで実施する設定 :
マニュアル「Hitachi Command Suite Tuning Manager - Agents」

1.2 http(s)経由で取得できる情報

API を利用して次に示す情報を取得できます。

- HTM - Agents が収集するパフォーマンスデータに対するアラート定義情報
HTM - Agents が収集するパフォーマンスデータに対して設定したしきい値の情報を取得できます。
- HTM - Agents が収集するパフォーマンスデータ
HTM - Agents が監視対象から収集するパフォーマンスデータを取得できます。
なお、取得できるデータは Collection Interval ごとに収集する履歴データおよび、時単位／日単位などの要約データです。リアルタイムデータについては取得できません。
また、デフォルト保持期間は、履歴データおよび各要約データの種類によって異なります。各データのデフォルト値については、マニュアル「Hitachi Command Suite Tuning Manager - Agents」を参照してください。

注意

Store データベースで運用しているエージェントからの情報の場合、履歴データのデフォルト保持期間は 48 時間ですが、要約データについては、取得できません。

保持期間を変更する方法についてはマニュアル「Hitachi Command Suite Tuning Manager 運用管理ガイド」を参照してください。

- API の構成情報
API のバージョン、Tuning Manager シリーズのバージョンなどの情報を取得できます。
- HTM - Agents 一覧
HTM - Agents のインスタンス情報、監視対象の情報の一覧を取得できます。
- パフォーマンスデータの取得に関連した HTM - Agents 情報のメタ情報、および API で扱うリソースのメタ情報
API を利用して取得する情報のメタ情報（属性名や型など）を取得できます。

1.3 http(s)経由で更新できる情報

API を利用して次に示す情報を更新できます。

- HTM - Agents が収集するパフォーマンスデータに対するアラート定義情報
- HTM - Agents 一覧の情報
 - パフォーマンスデータの取得の場合、HTM - Agents 一覧の情報は、定期的に自動で更新しています。更新間隔は変更できます。自動更新についての詳細は、マニュアル「Hitachi Command Suite Tuning Manager 運用管理ガイド」のエージェント情報のリフレッシュについて説明している個所を参照してください。
 - アラート機能の場合、手動で HTM - Agents 一覧の情報を更新する必要があります。

1.4 API で監視対象のしきい値情報にアクセスできる条件

監視対象のしきい値情報（アラート定義情報）にアクセスできる条件を次に示します。

アラート定義情報を参照するだけの場合

- 次のプログラムのバージョンがすべて v8.4 以降である。
 - Device Manager
 - HTM - Agents
- Tuning Manager server のライセンスが登録されている。
- アクセスするユーザーが Tuning Manager server の View 以上の権限を持っている。
- Hybrid Store で運用している。
- Tuning Manager server と Device Manager が連携している。
Tuning Manager server と Device Manager の連携については、マニュアル「Hitachi Command Suite システム構成ガイド」の関連製品と連携するために必要な設定を参照してください。

アラート定義情報を更新、変更する場合

- 参照するだけの場合の条件をすべて満たしている。
- アクセスするユーザーが Tuning Manager server の Admin 権限を持っている。

1.5 API でパフォーマンスデータにアクセスできる条件

API でパフォーマンスデータにアクセスできる条件を次に示します。

- Tuning Manager server のライセンスが登録されている。
- アクセスするユーザーが Tuning Manager server の View 以上の権限を持っている。
- Hybrid Store で運用^{※1}している。
- API で取得したいパフォーマンスデータのインスタンスが、HTM - Agents に 1 件以上存在する。
- API で取得したいパフォーマンスデータが、HTM - Agents によって定期収集されている。^{※2}

注※1

Store データベースで運用しているかつ Tuning Manager API の利用を有効化している場合でも、アクセスできます。

注※2

Store データベースで運用しているかつ Tuning Manager API の利用を有効化している場合は、API で取得したいパフォーマンスデータが稼働性能情報ファイルの出力対象レコードに設定されている必要があります。デフォルトでは、定期収集されているレコードが稼働性能情報ファイルに出力されます。

1.6 Tuning Manager API の最大リクエスト数

Tuning Manager API の最大リクエスト数を API の用途ごとに次に示します。

アラート機能で稼働監視する場合

API クライアントおよび `htmrest` コマンドを実行する CLI クライアントから **Device Manager** に対して同時に実行できる最大リクエスト数は 1 です。

同時に複数のリクエストを実行しないで、1つのリクエストに対してレスポンスが返却されてから、次のリクエストを実行してください。同時に複数のリクエストを実行すると、**Device Manager** の動作が一時的に遅くなることがあります。

パフォーマンスデータを取得する場合

API クライアントおよび `htmrest` コマンドを実行する CLI クライアントから **Tuning Manager server** に対して同時に実行できる最大リクエスト数は 40 です。ただし、**Tuning Manager server** から 1つのエージェントホストに対して、同時に実行できる最大リクエスト数は 8 です。

最大リクエスト数を超えると、エラーレスポンスを返却します。同時リクエスト数が上限を超えないように実行間隔を設定してください。

1.7 Tuning Manager API を利用するために必要なインストールとセットアップの流れ

Tuning Manager API を利用するために必要なインストールおよびセットアップの流れを説明します。

Tuning Manager API を有効化する **HTM - Agents** を非クラスタシステムで運用する場合には「[1.7.1 インストールとセットアップの流れ（非クラスタシステムで HTM - Agents を運用する場合）](#)」を、クラスタシステムで運用する場合には「[1.7.2 インストールとセットアップの流れ（クラスタシステムで HTM - Agents を運用する場合）](#)」を、それぞれ参照してください。

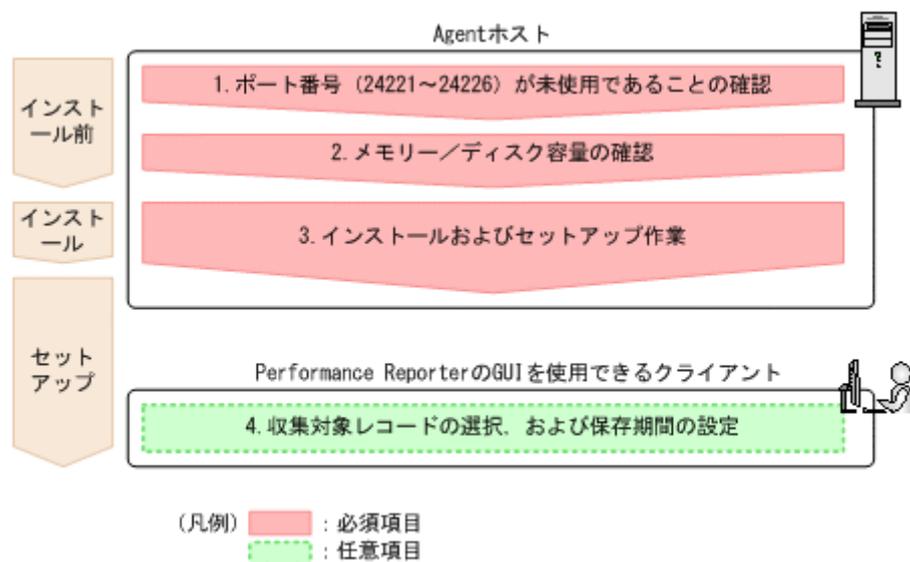
なお、ここでは、Tuning Manager API を利用する場合にだけ実施する作業項目について記載しています。

通常のインストールとセットアップについての詳細は、マニュアル「[Hitachi Command Suite Tuning Manager インストールガイド](#)」およびマニュアル「[Hitachi Command Suite Tuning Manager - Agents](#)」を参照してください。

1.7.1 インストールとセットアップの流れ（非クラスタシステムで HTM - Agents を運用する場合）

Tuning Manager API を有効化する HTM - Agents を非クラスタシステムで運用する場合、Tuning Manager API を利用するために必要なインストールおよびセットアップの流れについて説明します。

図 1-3 Tuning Manager API を利用するために必要なインストールとセットアップの流れ（非クラスタシステムで HTM - Agents を運用する場合）



注意：この流れ図に示す各作業項目の詳細については、次の表で参照先を確認してください。

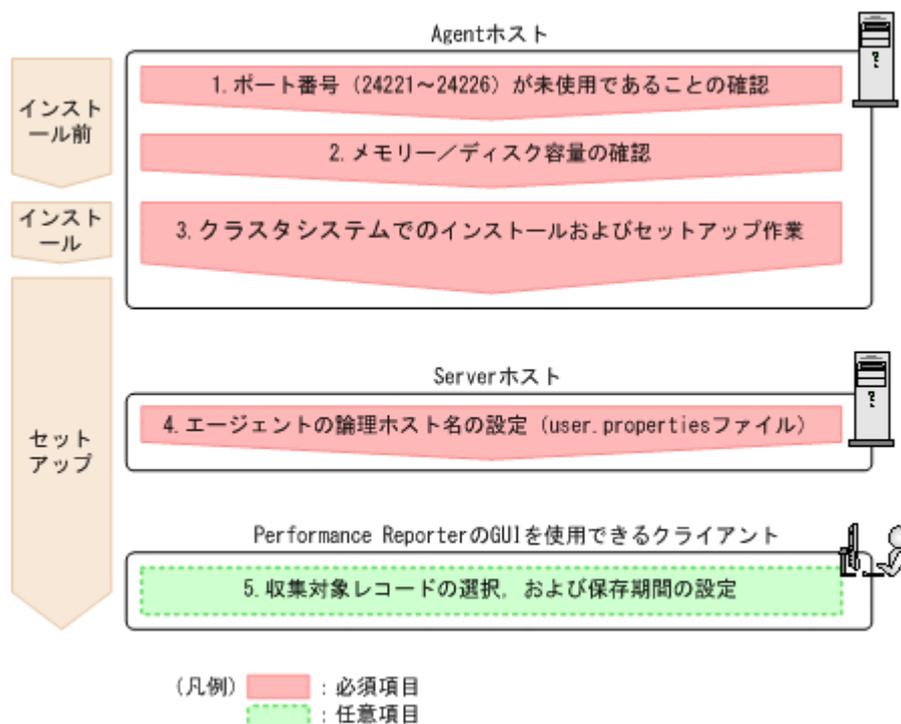
表 1-1 流れ図に示す作業項目の参照先一覧（非クラスタシステムで HTM - Agents を運用する場合）

流れ図上の番号/作業項目			参照マニュアル/参照先
1	ポート番号 (24221~24226) が未使用であることの確認	必須	マニュアル「Hitachi Command Suite Tuning Manager - Agents」 24221~24226 のポート番号を使用していた場合、次の参照先を確認の上、変更してください。 Tuning Manager API を利用する場合のポート番号の変更について説明している箇所
2	メモリー/ディスク容量の確認	必須	マニュアル「Hitachi Command Suite Tuning Manager - Agents」 システム見積もりについて説明している箇所
3	インストールおよびセットアップ作業	必須	マニュアル「Hitachi Command Suite Tuning Manager - Agents」 インストールおよびセットアップ手順について説明している箇所
4	収集対象レコードの選択、および保存期間の設定	任意	マニュアル「Hitachi Command Suite Tuning Manager 運用管理ガイド」 エージェントの管理と設定について説明している箇所

1.7.2 インストールとセットアップの流れ（クラスタシステムで HTM - Agents を運用する場合）

ここでは、Tuning Manager API を有効化する HTM - Agents をクラスタシステムで運用する場合、Tuning Manager API を利用するために必要なインストールおよびセットアップの流れについて説明します。

図 1-4 Tuning Manager API を利用するために必要なインストールとセットアップの流れ（クラスタシステムで HTM - Agents を運用する場合）



注意：この流れ図に示す各作業項目の詳細については、次の表で参照先を確認してください。

表 1-2 流れ図に示す作業項目の参照先一覧（クラスタシステムで HTM - Agents を運用する場合）

流れ図上の番号/作業項目			参照マニュアル/参照先
1	ポート番号 (24221~24226) が未使用であることの確認	必須	マニュアル「Hitachi Command Suite Tuning Manager - Agents」 24221~24226 のポート番号を使用していた場合、次の参照先を確認の上、変更してください。 Tuning Manager API を利用する場合のポート番号の変更について説明している箇所
2	メモリー/ディスク容量の確認	必須	マニュアル「Hitachi Command Suite Tuning Manager - Agents」 システム見積りについて説明している箇所
3	クラスタシステムでのインストールおよびセットアップ作業	必須	マニュアル「Hitachi Command Suite Tuning Manager - Agents」 クラスタシステムでのインストールおよびセットアップ手順について説明している箇所
4	エージェントの論理ホスト名設定 (user.properties ファイル)	必須	マニュアル「Hitachi Command Suite Tuning Manager 運用管理ガイド」

流れ図上の番号／作業項目		参照マニュアル／参照先	
			Tuning Manager API の利用を有効化しているエージェントの ホスト名を変更した場合の設定について説明している個所
5	収集対象レコードの選択, および保存期間の設定	任意	マニュアル「Hitachi Command Suite Tuning Manager 運用管 理ガイド」 エージェントの管理と設定について説明している個所

Tuning Manager API の仕様

この章では Tuning Manager API の仕様について説明します。

- 2.1 Tuning Manager API の共通仕様
- 2.2 サポートしている HTTP メソッド
- 2.3 リクエストメッセージ
- 2.4 セキュリティと認証
- 2.5 圧縮転送
- 2.6 Tuning Manager API のバージョン
- 2.7 共通オブジェクト
- 2.8 ステータスコード

2.1 Tuning Manager API の共通仕様

Tuning Manager API の共通仕様について説明します。

API で扱いたいリソースを指定するには、URI を使用します。次に示す URI の形式の<schema>から<version>までが基底 URI となり、<domain>以下が、相対 URI となります。リソースは<domain>以下の相対 URI に指定します。

URI の形式と、構成要素を示します。

形式

アラート機能で稼働監視する場合

```
<schema>://<host>:<port>/<application>/<version>/<domain>/<object-type>[/<AlertDefId>][?<query-string>]
```

または

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

パフォーマンスデータを取得する場合

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

または

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

構成要素

表 2-1 URI の構成要素

項目	説明
<schema>	プロトコル。 HTTP 「http」と HTTPS 「https」をサポートしています。
<host>	アラート機能で稼働監視する場合 API クライアントがアクセスする Device Manager のホスト名、または IP アドレス パフォーマンスデータを取得する場合 API クライアントがアクセスする Tuning Manager server のホスト名、または IP アドレス
<port>	アラート機能で稼働監視する場合 API クライアントがアクセスする Device Manager のポート番号 デフォルトは HTTP の場合は 22015、HTTPS の場合は 22016 です。 パフォーマンスデータを取得する場合 API クライアントがアクセスする Tuning Manager server のポート番号 デフォルトは HTTP の場合は 22015、HTTPS の場合は 22016 です。
<application>	アプリケーション名。 動作させる API の種類によって、指定するアプリケーション名が異なります。指定するアプリケーション名を次に示します。 <ul style="list-style-type: none">アラート機能で稼働監視する場合：DeviceManagerWebService/analyticsパフォーマンスデータを取得する場合：TuningManager
<version>	Tuning Manager API のバージョン。 API サービスの基底 URI となります。 「v<x>」（<x>は API バージョンのメジャーバージョン）の形式です。API バージョンが 1.<y>（<y>は API バージョンのマイナーバージョン。例えば、1.0、1.1、1.2 など）の場合は<version>は「v1」となり、API バージョンが 2.<y>（例えば、2.0、2.1、2.2 など）の場合は「v2」となります。

項目	説明
	Tuning Manager シリーズのバージョンと API のバージョンの対応については、「2.6」を参照してください。
<domain>	リソースのカテゴリ。* 次の 4 種類のドメインを提供します。 <ul style="list-style-type: none"> • configuration API の詳細情報 • services HTM - Agents のインスタンスに依存しないサービス情報 • objects HTM - Agents のアラート定義情報の設定に関する情報、およびインスタンス情報や監視対象から収集したパフォーマンスデータ • domain-types API で使用するリソースのメタデータ
<object-type>	リソース、サービスの種類。* 例: Objects URI の alert-defs や RAID_PD("<Agent 種別_レコード名>"の形式), Service URL の AgentManager
<AlertDefId>	アラート定義名。*
<action-identifier>	任意のアクションを表す識別子。* 例: Service URI の refresh
<query-string>	クエリ文字列。 アラート機能で稼働監視する場合 参照したいアラート定義を AlertDefName または Target で指定します。 パフォーマンスデータを取得する場合 <object-type>を絞り込む条件などを指定します。リソースごとに指定可能なクエリ文字列は異なります。クエリ文字列がアルファベットを含む場合、インスタンス名、性能情報の時間粒度および取得方法を除き、大文字小文字は区別されます。また、クエリ文字列同士の順序は問いません。 なお、各リソースで定義されていないクエリ文字列を指定した場合は以下の状態になります。 <ul style="list-style-type: none"> ◦ リソースがパフォーマンスデータの場合はリクエスト不正となります。 ◦ リソースがパフォーマンスデータ以外の場合は無視されます。 クエリ文字列の指定方法については、各 API の説明を参照してください。

注※

指定できるリソースの詳細については、「3.1 Tuning Manager API のリソース一覧」の「表 3-1 アラート機能のアラート定義情報を設定および運用する API」、 「4.1 Tuning Manager API のリソース一覧」の「表 4-1 Tuning Manager API のリソース一覧」、および「5.1 Tuning Manager API のリソース一覧」の「表 5-1 Tuning Manager API のリソース一覧」の URI を参照してください。

URI の指定例 (RAID_PI_LDS レコードのパフォーマンスデータを指定する場合)

なお、見やすくするために改行を入れていますが、実際指定する際には改行を入れてはいけません。

```
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
```

<domain>以下に指定できるリソースについては、「3.1 Tuning Manager API のリソース一覧」の「表 3-1 アラート機能のアラート定義情報を設定および運用する API」、 「4.1 Tuning Manager API

のリソース一覧」の「表 4-1 Tuning Manager API のリソース一覧」, および「5.1 Tuning Manager API のリソース一覧」の「表 5-1 Tuning Manager API のリソース一覧」を参照してください。

2.2 サポートしている HTTP メソッド

Tuning Manager API では HTTP が定義する次の操作メソッドをサポートしています。

- GET メソッド
- PUT メソッド
- POST メソッド
- DELETE メソッド

扱うリソースによって、使用できるメソッドは異なります。使用できるメソッドについては「3. アラート機能の API」, 「4. パフォーマンスデータを取得するための API」, または「5.1 Tuning Manager API のリソース一覧」のリソース一覧を参照してください。

2.3 リクエストメッセージ

Web クライアントから Web サーバに送信されるリクエストメッセージは、次の表に示す情報から構成されます。

表 2-2 リクエストメッセージの構成

項目		説明
リクエストヘッダー	リクエストライン	Web サーバにリクエストする情報を指定します。詳細は、「2.3.1」を参照してください。
	ヘッダー	Web ブラウザーで受信可能なデータのタイプや圧縮方法などを指定します。詳細は、「2.3.2」を参照してください。
	空白行	リクエストヘッダーの終わりを示します。
リクエストボディ		Web サーバに送信する情報を指定します。 GET メソッドの場合は空でなければいけません。

2.3.1 リクエストライン

リクエストラインは、次の表に示す項目から構成されます。

表 2-3 リクエストラインの構成

(a) メソッド	(b) スペース	(c) リクエストパス (URI)	(d) クエリ	(e) スペース	(f) HTTP バージョン
-------------	-------------	----------------------	------------	-------------	-------------------

リクエストラインを構成する各項目について説明します。項目(a)~(f)は、「表 2-3 リクエストラインの構成」と対応しています。

表 2-4 リクエストラインを構成する項目

項目	説明	例
(a)メソッド	リクエストのメソッド	GET
(b)スペース	区切りを示す空白文字 (1 バイト固定)	△ (半角スペース)
(c)リクエストパス (URI)	リソースを示すパス	/<application>/v1/objects/RAID_PD

項目	説明	例
(d)クエリ	リクエストに付加されたクエリ文字列	?hostName=agent_1&agentInstanceName=VSPG1000_instance
(e)スペース	区切りを示す空白文字 (1バイト固定)	△ (半角スペース)
(f)HTTPバージョン	HTTPのバージョンを示す文字列	HTTP/1.1

なお、Tuning Manager API の場合、リクエストラインの最大長は 8,190 バイトです。最大長を超えてリクエストラインを指定した場合、ステータスコード 414 (Request-URI Too Long) が返されます。

注意

リクエスト時に、リクエストパスおよびクエリ（「表 2-3 リクエストラインの構成」の項目(c) および(d)）の値が内部処理によって書き換えられ、リクエストラインが最大長 (8,190 バイト) を超える場合があります。その場合、リクエストパスおよびクエリの値を見直した上で、複数回に分けてリクエストしてください。

2.3.2 ヘッダー

指定可能なヘッダーを次の表に示します。なお、ヘッダーの種類の名称は大文字、小文字を区別しません。

表 2-5 Tuning Manager シリーズで指定可能なヘッダー

ヘッダーの種類	説明	指定要否
Authorization	ユーザー認証に使用します。認証についての詳細は、「2.4」を参照してください。	必須
Accept	受信可能なレスポンスデータのメディアタイプを指定します。	Accept ヘッダー自体を省略する場合だけ省略できます。省略した場合は、すべてのメディアタイプが受信可能であることを表す「*」が指定されたと見なします。 注意 Tuning Manager API では、Accept ヘッダーのメディアタイプを個別に指定する場合、「application/json」の指定を必須とし、さらにメタデータ取得の際は「application/schema+json」を、パフォーマンスデータを取得する際は「text/csv」を指定する必要があります。
Accept-Encoding	レスポンスを圧縮転送する場合に指定します。圧縮転送についての詳細は、「2.5」を参照してください。	任意
Content-Type	リクエストボディのメディアタイプを指定します。	GET メソッド使用時は指定不要です。指定した場合は無視されます。 PUT メソッドおよび POST メソッド使用時は、「application/json」の指定が必須です。
Content-Length	リクエストボディのサイズを 10 進数で指定します (単位: バイト)。	GET メソッド使用時は任意です。ただし指定する場合のサイズは 0 とします。0 以外を指定した場合には、Web サーバの仕様に従って処理されます。 PUT メソッドおよび POST メソッド使用時は、Content-Length、または Transfer-Encoding のどちらかの指定が必須です。

ヘッダーの種類	説明	指定要否
Transfer-Encoding	リクエストボディの転送コーディングを指定します。 HTTP1.1では「chunked」を指定できます。	GET メソッド使用時は任意です。ただし指定する場合のサイズは0とします。 PUT メソッドおよび POST メソッド使用時は、Content-Length, または Transfer-Encoding のどちらかの指定が必須です。

Accept ヘッダーについては、複数行で指定した場合でも1行で指定した場合と同じと見なされます。例を次に示します。

ヘッダーを複数行で指定した場合の例

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

ヘッダーを1行で指定した場合の例

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

また、Accept 以外のヘッダーは、1種類のヘッダーを複数行で指定した場合と1行で指定した場合とは同じと見なされません。Accept 以外のヘッダーを複数行で指定した場合の動作を次の表に示します。

表 2-6 複数行で指定した場合でも1行指定と同じと見なされないヘッダー

ヘッダーの種類	複数指定時の動作
Authorization	最初に指定したヘッダーを優先します。
Accept-Encoding	gzip, identity の順で優先します。
Content-Type	最初に指定したヘッダーを優先します。
Content-Length	Web サーバの動作に従います。
Transfer-Encoding	Web サーバの動作に従います。
Content-Length および Transfer-Encoding	これらの2種類のヘッダーが指定された場合、Transfer-Encoding が優先されます。

2.4 セキュリティと認証

API を使用するには、リクエストメッセージに Authorization ヘッダーを指定し認証方式と認証情報を送信する必要があります。サポートする認証方式は、Basic 認証です。標準の Basic 認証の指定方法に従って指定します。

Basic 認証を使用するためには、リクエストメッセージに Authorization ヘッダーを指定します。Authorization ヘッダーの内容には、認証方式"Basic"に続けて、ユーザー名とパスワードを":" (ASCII:0x3A)で連結した文字列を Base64 エンコードしたもの (basic token と呼びます) を指定します。Basic 認証での Authorization ヘッダーを次に示します。

Basic 認証でのリクエストメッセージの Authorization ヘッダー

```
Authorization: Basic <basic token>
```

http, および https のどちらも使用できますが、セキュリティの向上のため、https によるアクセスを推奨します。

2.5 圧縮転送

遠隔地からのアクセスなど、低スループットの回線を利用したアクセスや、一回のリクエストで取得するデータが大容量となるケースでは、コンテンツを圧縮して転送することで転送時間を短縮できます。一方、高スループットな回線で小容量のデータを取得するケースでは、コンテンツの圧縮処理がボトルネックとなるため、データの容量に応じて圧縮転送をしてください。

圧縮転送は HTTP プロトコルのバージョンが 1.1 の場合にだけ使用できます。サポートしているエンコーディング方式は、「gzip」および「identity」です。なお、「identity」の場合、圧縮はしません。

API クライアントは、リクエストメッセージに Accept-Encoding ヘッダーを付与してエンコーディング方式を通知します。API クライアントは、対応可能なエンコーディング方式を複数列挙した上で優先度を指定できます。

なお、Accept-Encoding ヘッダーで「*」を通知した場合は、「gzip」として処理されます。

「gzip」または「identity」による圧縮処理を要求するリクエストメッセージの Accept-Encoding ヘッダーを次に示します。

圧縮処理を要求するリクエストメッセージの Accept-Encoding ヘッダー

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

コンテンツの圧縮は、すべての URI に対して適用できますが、レスポンスボディのサイズが 100 バイトを超える場合にだけ圧縮されます。

コンテンツを圧縮した場合は、レスポンスメッセージに Content-Encoding ヘッダーおよび Vary ヘッダーが含まれます。Content-Encoding ヘッダーは、使用したエンコーディング方式をクライアントに通知するためのヘッダーです。また Vary ヘッダーは圧縮によってレスポンスボディが変化していることを通知するヘッダーです。ただし、エンコーディング方式が「identity」の場合は、レスポンスメッセージに Content-Encoding ヘッダーおよび Vary ヘッダーは含まれません。

「gzip」でコンテンツを圧縮した場合のレスポンスメッセージに含まれる Content-Encoding ヘッダーおよび Vary ヘッダーを次に示します。

コンテンツを圧縮した場合のレスポンスメッセージに含まれる Content-Encoding ヘッダーおよび Vary ヘッダー

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

2.6 Tuning Manager API のバージョン

Tuning Manager シリーズの製品のバージョンと、API のメジャーバージョンおよびマイナーバージョンの対応を次の表に示します。

表 2-7 Tuning Manager シリーズの製品のバージョンと API のバージョンの対応（アラート機能で稼働監視する場合）

Tuning Manager シリーズの製品のバージョン	API のバージョン (<メジャーバージョン>.<マイナーバージョン>.<リビジョン>の形式)
v8.2.1 より前	非サポート
v8.2.1 以降	1.0.0

表 2-8 Tuning Manager シリーズの製品のバージョンと API のバージョンの対応（パフォーマンスデータを取得する場合）

Tuning Manager シリーズの製品のバージョン	API のバージョン (<メジャーバージョン>.<マイナーバージョン>の形式)
v8.0.0 より前	非サポート
v8.0.0 以降	1.0
v8.1.3 以降	1.1
v8.2.1 以降	1.2
v8.4.1 以降	1.3

2.7 共通オブジェクト

個別の API に依存しない共通オブジェクトについて説明します。なお、共通オブジェクトを含む全 JSON オブジェクトのフィールドは順不同です。

2.7.1 Collection オブジェクト

オブジェクトの集合を返す際に、オブジェクトのコンテナとして用います。Collection オブジェクトのスキーマを次の表に示します。

表 2-9 Collection オブジェクトのスキーマ

変数名	型	説明
items	object の配列	任意のオブジェクトの集合。

Collection オブジェクトの例を次に示します。

Collection オブジェクトの例

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

2.7.2 Error オブジェクト

リクエストに対してエラーが発生した場合、「2.8 ステータスコード」に示すステータスコードとともに、次の表に示すスキーマで定義する Error オブジェクトが返されます。ただし、Job オブジェクトを返す API の場合は、エラーの発生個所によって Job オブジェクトが返されることがあります。

表 2-10 Error オブジェクトのスキーマ

変数名	型	説明
errorSource	string	API クライアントが送信した URI。
message	string	エラーメッセージ。 ログ出力する場合と同等のエラーメッセージです。
messageID	string	メッセージ ID。 ログ出力する場合と同等のメッセージ ID です。
actions	string 型の配列	API クライアントが取るべき対処。

Error オブジェクトに含まれるメッセージおよび対処は、英語で返されます。

Error オブジェクトの例を次に示します。

Error オブジェクトの例 (アラート機能の場合)

```
{"errorSource": "http://server_1:22015/DeviceManagerWebService/
analytics/v1/objects/alert-defs",
"message": "KAIC18813-E Failed to read the request body.",
"messageID": "KAIC18813",
"actions": ["Check the request body, and then try again."]}
```

Error オブジェクトの例 (パフォーマンスデータの取得の場合)

```
{"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."]}
```

2.7.3 Action オブジェクト

リソースまたはサービスのアクションを実行する際のパラメーター用のコンテナです。Action オブジェクトのスキーマを次の表に示します。

表 2-11 Action オブジェクトのスキーマ

変数名	型	説明
name	string	アクション名。
url	string	アクションを実行する URI。
method	string	アクション実行時の HTTP メソッド。
type	string	アクション実行時のリクエストの Content-Type ヘッダーに指定するメディアタイプ。
parameters	PrimitiveValue オブジェクト型の配列	アクションを実行時に必要なパラメーター。要素数と各要素の意味はアクションごと (API ごと) に定義されます。

2.7.4 PrimitiveValue オブジェクト

非オブジェクトである変数をオブジェクトとして扱うためのコンテナです。PrimitiveValue オブジェクトのスキーマを次の表に示します。

表 2-12 PrimitiveValue オブジェクトのスキーマ

変数名	型	説明
type	string	変数の型。
value	string	変数の値。

PrimitiveValue オブジェクトの例を次に示します。

PrimitiveValue オブジェクトの例

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

2.8 ステータスコード

レスポンスメッセージのステータスコードを次に表に示します。

表 2-13 レスポンスメッセージのステータスコード

ステータスコード	Name	内容
200	OK	リクエストが適切に処理された場合 なお、このステータスコードでもエラーが発生していることがあります。詳細は出力されるメッセージを確認してください。
201	Created	リクエストが適切に処理され新しいリソースが作成された場合 なお、このステータスコードでもエラーが発生していることがあります。詳細は出力されるメッセージを確認してください。
400	Bad Request	必須ヘッダーが省略された場合、クエリ文字列不正の場合、またはリクエストボディが不正だった場合
401	Unauthorized	Authorization ヘッダーが省略された場合、または認証に失敗した場合
403	Forbidden	権限が不足している場合
404	Not Found	要求したリソースが存在しない場合
405	Method Not Allowed	リソースに許可されていないメソッドを要求した場合
406	Not Acceptable	Accept ヘッダーや Accept-Encoding ヘッダーにサポートしていないメディアタイプが指定されている場合
414	Request-URI Too Long	リクエストラインが長過ぎる場合
415	Unsupported Media Type	メディアタイプの指定が不正だった場合
500	Internal Server Error	Tuning Manager server および HTM - Agents で内部エラーが発生した場合
503	Service Unavailable	Tuning Manager server および HTM - Agents の流量制限によって、一時的にアクセスができなくなった場合

アラート機能の API

この章では、アラート機能のアラート定義情報を設定および運用する API について、説明します。

アラート機能を使用する流れについては、「[A.1 アラート機能を設定・運用する流れ](#)」を参照してください。

なお、実行例には見やすくするために改行を入れています。

- [3.1 Tuning Manager API のリソース一覧](#)
- [3.2 アラート定義情報の追加](#)
- [3.3 アラート定義情報の削除](#)
- [3.4 アラート定義情報の参照](#)
- [3.5 アラート定義情報の同期実行時のパラメータの取得](#)
- [3.6 アラート定義情報の同期](#)
- [3.7 HTM - Agents 情報更新時のパラメーターの取得](#)
- [3.8 HTM - Agents 情報の更新](#)
- [3.9 HTM - Agents 情報取得時のパラメーターの取得](#)
- [3.10 HTM - Agents 情報の取得](#)
- [3.11 アラート定義情報設定時のオブジェクト](#)

3.1 Tuning Manager API のリソース一覧

アラート機能のアラート定義情報を設定および運用する API のリソース一覧を次の表に示します。
なお、リソースは URI で表記しています。

表 3-1 アラート機能のアラート定義情報を設定および運用する API

API の用途		リクエスト		参照先
		HTTP method	URI	
アラート定義情報を設定する API				
アラート定義情報の追加		POST	<code><application>/v1/objects/alert-defs</code>	3.2
アラート定義情報の削除		DELETE	<code><application>/v1/objects/alert-defs/<AlertDefId></code>	3.3
アラート定義情報の参照	1 件	GET	<code><application>/v1/objects/alert-defs/<AlertDefId></code>	3.4
	複数件	GET	<code><application>/v1/objects/alert-defs?<AlertDefQuery></code>	3.4
	全件	GET	<code><application>/v1/objects/alert-defs</code>	3.4
アラート定義情報の同期時のパラメーターの取得		GET	<code><application>/v1/services/alert-def-manager/actions/sync</code>	3.5
アラート定義情報の同期		PUT	<code><application>/v1/services/alert-def-manager/actions/sync/invoke</code>	3.6
アラート機能を運用する API				
HTM - Agents 情報更新時のパラメーターの取得		GET	<code><application>/v1/services/agent-manager/actions/refresh</code>	3.7
HTM - Agents 情報の更新		PUT	<code><application>/v1/services/agent-manager/actions/refresh/invoke</code>	3.8
HTM - Agents 情報取得時のパラメーターの取得		GET	<code><application>/v1/services/agent-manager/actions/list</code>	3.9
HTM - Agents 情報の取得		PUT	<code><application>/v1/services/agent-manager/actions/list/invoke</code>	3.10

3.2 アラート定義情報の追加

アラート定義情報を新規に追加します。

HTTP リクエスト構文 (URI)

POST `<scheme>://<host>:<port>/<application>/<version>/objects/alert-defs`

リクエストの構造

リクエストにはアラート定義型オブジェクトを指定してください。

アラート定義型オブジェクトについては、「3.11 アラート定義情報設定時のオブジェクト」および「A.2 アラート定義情報の作成手順」を参照してください。

レスポンスの構造

レスポンスボディの構造を次に示します。なお、Error オブジェクトおよび cause 要素は、エラーが発生している場合に含まれます。

```
{
  "state": "...",
  "affectedResources": [
    "...",
  ],
  "error": {
    "errorSource": "...",
    "message": "...",
    "messageID": "...",
    "actions": [
      "...",
    ]
  },
  "result": [
    {
      "distributionResult": "...",
      "agent": {
        "agentType": "...",
        "hostName": "...",
        "instanceName": "...",
        "model": "...",
        "serialNumber": "...",
        "smuIp": "...",
        "destIp": "...",
        "detourIp": "..."
      },
      "cause": {
        "errorID": "...",
        "errorDescription": "..."
      }
    }
  ],
  "resultType": "AlertDefDistributionResult"
}
```

表 3-2 アラート定義情報追加時のスキーマ

変数名	型	説明
state	string	処理結果の状態 <ul style="list-style-type: none"> • Succeeded: 成功 • Failed: 失敗 • Distribution Failed: 失敗 (エージェントへのアラート定義配布) Distribution Failed の場合、返却された cause 要素の内容を確認して、失敗の要因を取り除いてから、アラート定義情報の同期を実行してください。
affectedResources	string	追加したアラート定義情報を参照するための URI
error ^{※1}	—	エラー時の情報 詳細については、「2.7.2」を参照してください。
— errorSource	string	API クライアントが送信した URI
— message	string	エラーメッセージ
— messageID	string	メッセージ ID
— actions	string 型の配 列	API クライアントが取るべき対処
result	—	定義追加の結果情報
— distributionResult	string	追加の状態 <ul style="list-style-type: none"> • Succeeded: 成功

変数名	型	説明
		・ Failed: 失敗
agent	—	監視対象の情報
— agentType	string	エージェントの種別 ・ HTM - Agent for RAID の場合: RAID ・ HTM - Agent for NAS の場合: NAS
hostName	string	エージェントのホスト名
instanceName	string	エージェントのインスタンス名
model	string	ストレージシステムの機種名
serialNumber	string	ストレージシステムのシリアル番号
smuIp	string	外部 NAS Manager を特定するための情報 外部 NAS Manager で管理される NAS Platform のインスタンスを作成するときに指定した SMU/NAS Manager IP Address が返却されます。
destIp	string	Hitachi Virtual File Platform ^{※2} を特定するための情報 Hitachi Virtual File Platform ^{※2} のインスタンスを作成するときに指定した destination_address が返却されます。
detourIp	string	Hitachi Virtual File Platform ^{※2} を特定するための情報 Hitachi Virtual File Platform ^{※2} のインスタンスを作成するときに指定した detour_address が返却されます。
cause ^{※1}	—	エラー時の情報
— errorID	string	メッセージ ID
errorDescription	string	エラーメッセージ
resultType	string	次の値が返却されます。 AlertDefDistributionResult

注※1

Succeeded の場合は返却されません。

注※2

次のファイルサーバを総称して、「**Hitachi Virtual File Platform**」と呼びます。

- Hitachi Virtual File Platform
- Hitachi Capacity Optimization
- Hitachi Essential NAS Platform

実行例（アラート定義情報の追加）

curl コマンドを使用したリクエストメッセージ:

```
curl -v -H "Content-Type: application/json" -u system:manager -d
@AlertVSPG1000_01.json -X POST http://server_1:22015/
DeviceManagerWebService/analytics/v1/objects/alert-defs
```

リクエストに対する実行結果

リクエストヘッダー:

```
POST /DeviceManagerWebService/analytics/v1/objects/alert-defs HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/
3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22015
```

Accept: */*
Content-Type: application/json
Content-Length: 1024

レスポンスヘッダー:

HTTP/1.1 201 Created
Date: Tue, 20 Oct 2015 07:07:40 GMT
Server: Cosminexus HTTP Server
Location: http://server_1:22015/DeviceManagerWebService/analytics/v1/objects/alert-defs/AlertVSPG1000_01
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json;charset=UTF-8

レスポンスボディ:

```
{
  "state": "Succeeded",
  "affectedResources": [
    "http://server_1:22015/DeviceManagerWebService/analytics/v1/objects/alert-defs/AlertVSPG1000_01"
  ],
  "result": [
    {
      "distributionResult": "Succeeded",
      "agent": {
        "agentType": "RAID",
        "hostName": "agent_1",
        "instanceName": "VSPG1000_instance",
        "model": "VSP G1000",
        "serialNumber": "10182"
      }
    }
  ],
  "resultType": "AlertDefDistributionResult"
}
```

リクエスト時に読み込ませたファイルの内容: (AlertVSPG1000_01.json)

```
{
  "alertDefName": "AlertVSPG1000_01",
  "targets": [
    {
      "targetID": "target1",
      "targetDef": [
        {
          "type": "model",
          "key": {
            "model": "VSP G1000"
          }
        }
      ]
    }
  ],
  "notices": [
    {
      "noticeID": "notice1",
      "emailNotification": [
        "warning", "caution", "normal"
      ],
      "emails": [
        {
          "addresses": [
            "alert@example.com"
          ],
          "subject": "HTnM Alert Mail",
          "body": "Alert Mail"
        }
      ]
    }
  ]
}
```

```

    ]
  }
],
"triggers": [
  {
    "triggerID": "trigger1",
    "isActive": "Y",
    "monitorAlways": "Y",
    "alertEveryTime": "Y",
    "evaluateAllData": "Y",
    "condition": {
      "warning": "%PI_LDS.READ_IO_COUNT% > '3000'",
      "caution": "%PI_LDS.READ_IO_COUNT% > '2000'"
    }
  }
],
"definitions": [
  {
    "targetIDs": [
      "target1"
    ],
    "noticeIDs": [
      "notice1"
    ],
    "triggerIDs": [
      "trigger1"
    ]
  }
],
"formatVersion": "1.0"
}

```

3.3 アラート定義情報の削除

追加したアラート定義情報を削除します。

HTTP リクエスト構文 (URI)

DELETE <scheme>://<host>:<port>/<application>/<version>/objects/alert-
defs/<AlertDefId>

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

レスポンスボディの構造を次に示します。なお、Error オブジェクトおよび cause 要素は、エラーが発生している場合に含まれます。

```

{
  "state": "...",
  "error": {
    "errorSource": "...",
    "message": "...",
    "messageID": "...",
    "actions": [
      "..."
    ]
  },
  "result": [
    {
      "deleteResult": "Succeeded",
      "agent": {
        "agentType": "...",
        "hostName": "...",
        "instanceName": "...",

```

```

        "model": "...",
        "serialNumber": "...",
        "smuIp": "...",
        "destIp": "...",
        "detourIp": "...",
    },
    "cause": {
        "errorID": "...",
        "errorDescription": "..."
    }
}
],
"resultType": "AlertDefDeleteResult"
}

```

表 3-3 アラート定義情報削除時のスキーマ

変数名	型	説明	
state	string	処理結果の状態 <ul style="list-style-type: none"> • Succeeded: 成功 • Failed: 失敗 • Distribution Failed: 失敗 (エージェントからのアラート定義情報削除) Distribution Failed の場合、返却された cause 要素の内容を確認して、失敗の要因を取り除いてから、アラート定義情報の同期を実行してください。	
error※1	—	エラー時の情報 詳細については、「2.7.2」を参照してください。	
—	errorSource	string	API クライアントが送信した URI
—	message	string	エラーメッセージ
—	messageID	string	メッセージ ID
—	actions	string 型の配 列	API クライアントが取るべき対処
result	—	定義削除の結果情報	
—	deleteResult	string	削除の状態 <ul style="list-style-type: none"> • Succeeded: 成功 • Failed: 失敗
—	agent	—	監視対象の情報
—	agentType	string	エージェントの種別 <ul style="list-style-type: none"> • HTM - Agent for RAID の場合: RAID • HTM - Agent for NAS の場合: NAS
—	hostName	string	エージェントのホスト名
—	instanceName	string	エージェントのインスタンス名
—	model	string	ストレージシステムの機種名
—	serialNumber	string	ストレージシステムのシリアル番号
—	smuIp	string	外部 NAS Manager を特定するための情報 外部 NAS Manager で管理される NAS Platform のインスタンスを作成するときに指定した SMU/NAS Manager IP Address が返却されます。
—	destIp	string	Hitachi Virtual File Platform※2 を特定するための情報 Hitachi Virtual File Platform※2 のインスタンスを作成するときに指定した destination_address が返却されます。
—	detourIp	string	Hitachi Virtual File Platform※2 を特定するための情報 Hitachi Virtual File Platform※2 のインスタンスを作成するときに指定した detour_address が返却されます。

変数名	型	説明
cause※1	—	エラー時の情報
— errorID	string	メッセージ ID
— errorDescription	string	エラーメッセージ
resultType	string	次の値が返却されます。 AlertDefDeleteResult

注※1

Succeeded の場合は返却されません。

注※2

次のファイルサーバを総称して、「**Hitachi Virtual File Platform**」と呼びます。

- Hitachi Virtual File Platform
- Hitachi Capacity Optimization
- Hitachi Essential NAS Platform

実行例（アラート定義情報の削除）

curl コマンドを使用したリクエストメッセージ：

```
curl -v -H "Content-Type: application/json" -u system:manager -X DELETE
http://server_1:22015/DeviceManagerWebService/analytics/v1/objects/alert-
defs/AlertVSPG1000_04
```

リクエストに対する実行結果／

リクエストヘッダー：

```
DELETE /DeviceManagerWebService/analytics/v1/objects/alert-defs/
AlertVSPG1000_04 HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/
3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22015
Accept: */*
Content-Type: application/json
```

レスポンスヘッダー：

```
HTTP/1.1 200 OK
Date: Tue, 20 Oct 2015 07:07:47 GMT
Server: Cosminexus HTTP Server
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json; charset=UTF-8
```

レスポンスボディ：

```
{
  "state": "Succeeded",
  "result": [
    {
      "deleteResult": "Succeeded",
      "agent": {
        "agentType": "RAID",
        "hostName": "agent_1",
        "instanceName": "VSPG1000_instance",

```

```

        "model": "VSP G1000",
        "serialNumber": "10182"
    }
}
],
"resultType": "AlertDefDeleteResult"
}

```

3.4 アラート定義情報の参照

追加したアラート定義情報を参照します。

HTTP リクエスト構文 (URI)

- 参照したいアラート定義が 1 件の場合

参照したいアラート定義のアラート定義名を指定します。

```
GET <scheme>://<host>:<port>/<application>/<version>/objects/alert-
defs/<AlertDefId>
```

- 参照したいアラート定義が複数件の場合

参照したいアラート定義をクエリ文字列で指定します。アラート定義名だけがレスポンスとして返却されます。

```
GET <scheme>://<host>:<port>/<application>/<version>/objects/alert-
defs?<query-string>
```

- 参照したいアラート定義が全件の場合

アラート定義名だけがレスポンスとして返却されます。

```
GET <scheme>://<host>:<port>/<application>/<version>/objects/alert-
defs
```

クエリ文字列

参照したいアラート定義が複数件の場合、条件に合わせて AlertDefName または Target のどちらかのキーを指定してください。

アラート定義名で指定する場合

クエリ文字列の指定例：

```
AlertDefName=Alert1%1FAlert2
```

「Alert1」と「Alert2」の 2 つのアラート定義を指定しています。

表 3-4 アラート定義取得時に指定可能なクエリ文字列のキーと値 (アラート定義名の場合)

キー	値	
	値域	概要
AlertDefName	1~127 バイト	アラート定義名を指定します。 複数指定したい場合は、キーと値の組み合わせを複数指定するか、値を%1Fで区切って指定してください。

監視対象で指定する場合

クエリ文字列の指定例：

```
Target=VSP%20G1000%1F12345
```

対象のストレージシステムが「VSP G1000」、かつ、シリアル番号が「12345」のアラート定義を指定しています。

表 3-5 アラート定義取得時に指定可能なクエリ文字列のキーと値（監視対象の場合）

キー	値	
	値域	概要
Target	1~1,024 バイト	取得したいアラート定義を絞り込むための文字列を指定します。絞り込み条件としては、target 型の targetDef 型の変数 key に定義する情報型の値を指定します。 複数の条件で絞り込む場合は、キーと値の組み合わせを複数指定するか、値を%1Fで区切って指定します。

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

参照したいアラート定義が 1 件の場合

```
{
  "alertDefName": "<アラート定義名>",
  <アラート定義型オブジェクト>
}
```

アラート定義型オブジェクトについては、「3.11 アラート定義情報設定時のオブジェクト」および「A.2 アラート定義情報の作成手順」を参照してください。

参照したいアラート定義が複数件または全件の場合

```
{
  "data": [
    {
      "alertDefName": "<アラート定義名>"
    },
    {
      "alertDefName": "<アラート定義名>"
    },
    {
      "alertDefName": "<アラート定義名>"
    }
  ],
  :
}
```

実行例 1（参照したいアラート定義が 1 件）

curl コマンドを使用したリクエストメッセージ：

```
curl -v -H "Content-Type: application/json" -u system:manager -X GET
http://server_1:22015/DeviceManagerWebService/analytics/v1/objects/alert-
defs/AlertVSPG1000_01
```

リクエストに対する実行結果／

リクエストヘッダー：

```
GET /DeviceManagerWebService/analytics/v1/objects/alert-defs/
AlertVSPG1000_01 HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/
3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22015
Accept: */*
Content-Type: application/json
```

レスポンスヘッダー:

```
HTTP/1.1 200 OK
Date: Tue, 20 Oct 2015 07:18:29 GMT
Server: Cosminexus HTTP Server
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json;charset=UTF-8
```

レスポンスボディ:

```
{
  "alertDefName": "AlertVSPG1000_01",
  "targets": [
    {
      "targetID": "target1",
      "targetDef": [
        {
          "type": "model",
          "key": {
            "model": "VSP G1000"
          },
          "filters": null
        }
      ]
    }
  ],
  "notices": [
    {
      "noticeID": "noticel",
      "emailNotification": [
        "warning",
        "caution",
        "normal"
      ],
      "emails": [
        {
          "addresses": [
            "alert@example.com"
          ],
          "subject": "HTnM Alert Mail",
          "body": "Alert Mail"
        }
      ]
    }
  ],
  "triggers": [
    {
      "triggerID": "trigger1",
      "isActive": "Y",
      "monitorAlways": "Y",
      "monitoringTime": null,
      "alertEveryTime": "Y",
      "evaluateAllData": "Y",
      "evaluatePoR": "N",
      "condition": {
        "targetOfPoR": null,
        "warning": "%PI_LDS.READ_IO_COUNT% > '3000'",
        "caution": "%PI_LDS.READ_IO_COUNT% > '2000'"
      },
      "frequency": null
    }
  ],
  "definitions": [
    {
      "targetIDs": [
        "target1"
      ],
      "noticeIDs": [
        "noticel"
      ]
    }
  ]
}
```

```

        ],
        "triggerIDs": [
            "trigger1"
        ]
    }
],
"formatVersion": "1.0"
}

```

実行例 2 (クエリ指定 (アラート定義名指定))

curl コマンドを使用したリクエストメッセージ:

```

curl -v -H "Content-Type: application/json" -u system:manager -X GET
http://server_1:22015/DeviceManagerWebService/analytics/v1/objects/alert-
defs?AlertDefName=AlertVSPG1000_01

```

リクエストに対する実行結果

リクエストヘッダー:

```

GET /DeviceManagerWebService/analytics/v1/objects/alert-defs?
AlertDefName=AlertVSPG1000_01 HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/
3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22015
Accept: */*
Content-Type: application/json

```

レスポンスヘッダー:

```

HTTP/1.1 200 OK
Date: Tue, 20 Oct 2015 07:18:30 GMT
Server: Cosminexus HTTP Server
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json; charset=UTF-8

```

レスポンスボディ:

```

{
  "data": [
    {
      "alertDefName": "AlertVSPG1000_01"
    }
  ]
}

```

実行例 3 (クエリ指定 (Target 指定))

curl コマンドを使用したリクエストメッセージ:

```

curl -v -H "Content-Type: application/json" -u system:manager -X GET
http://server_1:22015/DeviceManagerWebService/analytics/v1/objects/alert-
defs?Target=VSP%20G1000

```

リクエストに対する実行結果

リクエストヘッダー:

```

GET /DeviceManagerWebService/analytics/v1/objects/alert-defs?Target=VSP

```

```
%20G1000 HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/
3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22015
Accept: */*
Content-Type: application/json
```

レスポンスヘッダー:

```
HTTP/1.1 200 OK
Date: Tue, 20 Oct 2015 07:18:30 GMT
Server: Cosminexus HTTP Server
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json; charset=UTF-8
```

レスポンスボディ:

```
{
  "data": [
    {
      "alertDefName": "AlertVSPG1000_01"
    },
    {
      "alertDefName": "AlertVSPG1000_02"
    },
    {
      "alertDefName": "AlertVSPG1000_03"
    }
  ]
}
```

実行例 4 (参照したいアラート定義が全件)

curl コマンドを使用したリクエストメッセージ:

```
curl -v -H "Content-Type: application/json" -u system:manager -X GET
http://server_1:22015/DeviceManagerWebService/analytics/v1/objects/alert-
defs
```

リクエストに対する実行結果

リクエストヘッダー:

```
GET /DeviceManagerWebService/analytics/v1/objects/alert-defs HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/
3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22015
Accept: */*
Content-Type: application/json
```

レスポンスヘッダー:

```
HTTP/1.1 200 OK
Date: Tue, 20 Oct 2015 07:18:29 GMT
Server: Cosminexus HTTP Server
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json; charset=UTF-8
```

レスポンスボディ:

```

{
  "data": [
    {
      "alertDefName": "AlertVSPG1000_01"
    },
    {
      "alertDefName": "AlertVSPG1000_02"
    },
    {
      "alertDefName": "AlertVSPG1000_03"
    },
    {
      "alertDefName": "AlertUSPV_01"
    },
    {
      "alertDefName": "AlertUSPV_02"
    },
    {
      "alertDefName": "AlertUSPV_03"
    },
    {
      "alertDefName": "AlertVSP_01"
    },
    {
      "alertDefName": "AlertVSP_02"
    },
    {
      "alertDefName": "AlertVSP_03"
    }
  ]
}

```

3.5 アラート定義情報の同期実行時のパラメータの取得

Device Manager ホストと HTM - Agents ホストのアラート定義情報を同期する API のパラメータを取得します。

HTTP リクエスト構文 (URI)

GET <scheme>://<host>:<port>/<application>/<version>/services/alert-def-manager/actions/sync

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

```

{
  "name": "sync",
  "url": "<scheme>://<host>:<port>/<application>/<version>/services/alert-def-manager/actions/sync/invoke",
  "method": "PUT",
  "type": "application/json",
  "parameters": {
    "agentType": "<エージェント種別>",
    "hostName": "<エージェントホスト名>",
    "instanceName": "<エージェントインスタンス名>"
  }
}

```

実行例 (アラート定義情報の同期実行時のパラメータの取得)

curl コマンドを使用したリクエストメッセージ:

```
curl -v -H "Content-Type: application/json" -u system:manager -X GET
http://server_1:22015/DeviceManagerWebService/analytics/v1/services/
alert-def-manager/actions/sync
```

リクエストに対する実行結果

リクエストヘッダー:

```
GET /DeviceManagerWebService/analytics/v1/services/alert-def-manager/
actions/sync HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/
3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22015
Accept: */*
Content-Type: application/json
```

レスポンスヘッダー:

```
HTTP/1.1 200 OK
Date: Tue, 20 Oct 2015 07:34:13 GMT
Server: Cosminexus HTTP Server
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json; charset=UTF-8
```

レスポンスボディ:

```
{
  "name": "sync",
  "url": "http://server_1:22015/DeviceManagerWebService/analytics/v1/
services/alert-def-manager/actions/sync/invoke",
  "method": "PUT",
  "type": "application/json",
  "parameters": {
    "agentType": "all",
    "hostName": null,
    "instanceName": null,
    "formatVersion": "1.0"
  }
}
```

3.6 アラート定義情報の同期

Device Manager ホストと HTM - Agents ホストのアラート定義情報を同期します。Device Manager が管理しているアラート定義情報と HTM - Agents が管理しているアラート定義情報を一致させます。

HTTP リクエスト構文 (URI)

```
PUT <scheme>://<host>:<port>/<application>/<version>/services/alert-def-
manager/actions/sync/invoke
```

リクエストの構造

リクエストボディの構造を次に示します。

```
{
  "agentType" : "<エージェント種別>",
  "hostName" : "<ホスト名>",
```

```

    "instanceName" : "<インスタンス名>"
}

```

表 3-6 アラート定義同期時のリクエストボディのスキーマ

変数名	型	指定要否	説明
agentType	string	必須	エージェント種別 <ul style="list-style-type: none"> HTM - Agent for RAID の場合 : raid HTM - Agent for NAS の場合 : nas すべてのエージェント種別の場合 : all
hostName	string	任意	エージェントのホスト名 省略した場合は、すべてのホストが対象となります。
instanceName	string	任意	エージェントインスタンス名 省略した場合はすべてのエージェントインスタンスが対象となります。

レスポンスの構造

レスポンスボディの構造を次に示します。

```

{
  "state": "...",
  "error": {
    "errorSource": "...",
    "message": "...",
    "messageID": "...",
    "actions": [
      "..."
    ]
  },
  "result": [
    {
      "syncResult": "...",
      "agent": {
        "agentType": "...",
        "hostName": "...",
        "instanceName": "...",
        "model": "...",
        "serialNumber": "...",
        "smuIp": "...",
        "destIp": "...",
        "detourIp": "..."
      },
      "syncedAlertDef": [
        "...",
        "...",
        :
      ]
    },
    :
  ],
  "resultType": "AlertDefSyncResult"
}

```

表 3-7 アラート定義同期時のレスポンスボディのスキーマ

変数名	型	説明
state	string	処理結果の状態 <ul style="list-style-type: none"> Succeeded : 成功 Failed : 失敗 Distribution Failed : 失敗 (エージェントとの同期) Distribution Failed の場合、返却された cause 要素の内容を確認して、失敗の要因を取り除いてから、アラート定義情報の同期を実行してください。
error ^{※1}	—	エラー時の情報 詳細については、「2.7.2」を参照してください。

変数名	型	説明	
—	errorSource	string	API クライアントが送信した URI
	message	string	エラーメッセージ
	messageID	string	メッセージ ID
	actions	string 型の配列	API クライアントが取るべき対処
	result	—	定義同期の結果情報
—	syncResult	string	同期の状態 <ul style="list-style-type: none"> • Succeeded : 成功 • Failed : 失敗
	agent	—	監視対象の情報
—	agentType	string	エージェントの種類 <ul style="list-style-type: none"> • HTM - Agent for RAID の場合 : RAID • HTM - Agent for NAS の場合 : NAS
	hostName	string	エージェントのホスト名
	instanceName	string	エージェントのインスタンス名
	model	string	ストレージシステムの機種名
	serialNumber	string	ストレージシステムのシリアル番号
	smuIp	string	外部 NAS Manager を特定するための情報 外部 NAS Manager で管理される NAS Platform のインスタンスを作成するときに指定した SMU/NAS Manager IP Address が返却されます。
	destIp	string	Hitachi Virtual File Platform ^{※2} を特定するための情報 Hitachi Virtual File Platform ^{※2} のインスタンスを作成するときに指定した destination_address が返却されます。
	detourIp	string	Hitachi Virtual File Platform ^{※2} を特定するための情報 Hitachi Virtual File Platform ^{※2} のインスタンスを作成するときに指定した detour_address が返却されます。
	syncedAlertDef	string 型の配列	同期されたアラート定義名 同期時点でのエージェントに適用されたすべてのアラート定義名が返却されます。
	cause ^{※1}	—	エラー時の情報
—	errorID	string	メッセージ ID
	errorDescription	string	エラーメッセージ
	resultType	string	次の値が返却されます。 AlertDefSyncResult

注※1

Succeeded の場合は返却されません。

注※2

次のファイルサーバを総称して、「Hitachi Virtual File Platform」と呼びます。

- Hitachi Virtual File Platform
- Hitachi Capacity Optimization
- Hitachi Essential NAS Platform

実行例（アラート定義情報の同期）

curl コマンドを使用したリクエストメッセージ：

```
curl -v -H "Content-type: application/json" -u system:manager -d
'{"agentType":"all"}' -X PUT http://server_1:22015/
DeviceManagerWebService/analytics/v1/services/alert-def-manager/actions/
sync/invoke
```

リクエストに対する実行結果

リクエストヘッダー:

```
PUT /DeviceManagerWebService/analytics/v1/services/alert-def-manager/
actions/sync/invoke HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/
3.14.3.0 zlib/1.2.3 libidn/1.18 libssh2/1.4.2
Host: server_1:22015
Accept: */*
Content-type: application/json
Content-Length: 19
```

レスポンスヘッダー:

```
HTTP/1.1 200 OK
Date: Tue, 20 Oct 2015 07:36:31 GMT
Server: Cosminexus HTTP Server
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json; charset=UTF-8
```

レスポンスボディ:

```
{
  "state": "Succeeded",
  "result": [
    {
      "syncResult": "Succeeded",
      "agent": {
        "agentType": "RAID",
        "hostName": "agent_1",
        "instanceName": "VSPG1000_instance",
        "model": "VSP G1000",
        "serialNumber": "10182"
      },
      "syncedAlertDef": [
        "AlertVSPG1000_01",
        "AlertVSPG1000_02",
        "AlertVSPG1000_03"
      ]
    },
    {
      "syncResult": "Succeeded",
      "agent": {
        "agentType": "RAID",
        "hostName": "agent_1",
        "instanceName": "VSP_instance",
        "model": "VSP",
        "serialNumber": "53039"
      },
      "syncedAlertDef": [
        "AlertVSP_01",
        "AlertVSP_02",
        "AlertVSP_03"
      ]
    },
    {
      "syncResult": "Succeeded",
      "agent": {
        "agentType": "RAID",
```

```

        "hostName": "agent_1",
        "instanceName": "USPV_instance",
        "model": "USP V",
        "serialNumber": "10007"
    },
    "syncedAlertDef": [
        "AlertUSPV_01",
        "AlertUSPV_02",
        "AlertUSPV_03"
    ]
}
],
"resultType": "AlertDefSyncResult"
}

```

3.7 HTM - Agents 情報更新時のパラメーターの取得

HTM - Agents 情報を更新する API のパラメータを取得します。

HTTP リクエスト構文 (URI)

GET <scheme>://<host>:<port>/<application>/<version>/services/agent-manager/actions/refresh

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

```

{
  "name": "refresh",
  "url": "<scheme>://<host>:<port>/<application>/<version>/services/agent-manager/actions/refresh/invoke",
  "method": "PUT",
  "type": "application/json",
  "parameters": {
    "agentType": "<エージェント種別>"
  }
}

```

実行例 (HTM - Agents 情報更新時のパラメーターの取得)

curl コマンドを使用したリクエストメッセージ:

```

curl -v -H "Content-Type: application/json" -u system:manager -X GET
http://server_1:22015/DeviceManagerWebService/analytics/v1/services/agent-manager/actions/refresh

```

リクエストに対する実行結果

リクエストヘッダー:

```

GET /DeviceManagerWebService/analytics/v1/services/agent-manager/actions/refresh HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22015
Accept: */*
Content-Type: application/json

```

レスポンスヘッダー:

```

HTTP/1.1 200 OK
Date: Tue, 20 Oct 2015 07:34:14 GMT
Server: Cosminexus HTTP Server
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json;charset=UTF-8

```

レスポンスボディ :

```

{
  "name": "refresh",
  "url": "http://server_1:22015/DeviceManagerWebService/analytics/v1/
services/agent-manager/actions/refresh/invoke",
  "method": "PUT",
  "type": "application/json",
  "parameters": {
    "agentType": "all",
    "formatVersion": "1.0"
  }
}

```

3.8 HTM - Agents 情報の更新

HTM - Agents 情報を更新します。

HTTP リクエスト構文 (URI)

PUT <scheme>://<host>:<port>/<application>/<version>/services/agent-manager/actions/refresh/invoke

リクエストの構造

リクエストボディの構造を次に示します。

```

{
  "agentType" : "<エージェント種別>"
}

```

表 3-8 情報の更新時のリクエストボディのスキーマ

変数名	型	指定要否	説明
agentType	string	必須	エージェント種別 <ul style="list-style-type: none"> HTM - Agent for RAID の場合 : raid HTM - Agent for NAS の場合 : nas すべてのエージェント種別の場合 : all

レスポンスの構造

レスポンスボディの構造を次に示します。

```

{
  "state": "...",
  "error": {
    "errorSource": "...",
    "message": "...",
    "messageID": "...",
    "actions": [
      "..."
    ]
  }
}

```

表 3-9 情報の更新時のレスポンスボディのスキーマ

変数名	型	説明	
state	string	処理結果の状態 <ul style="list-style-type: none"> Succeeded : 成功 Failed : 失敗 	
error*	—	エラー時の情報 詳細については、「2.7.2」を参照してください。	
—	errorSource	string	API クライアントが送信した URI
	message	string	エラーメッセージ
	messageID	string	メッセージ ID
	actions	string	API クライアントが取るべき対処

注※

Succeeded の場合は返却されません。

実行例 (HTM - Agents 情報の更新)

curl コマンドを使用したリクエストメッセージ:

```
curl -v -H "Content-Type: application/json" -d '{"agentType":"all"}' -u
system:manager -X PUT http://server_1:22015/DeviceManagerWebService/
analytics/v1/services/agent-manager/actions/refresh/invoke
```

リクエストに対する実行結果

リクエストヘッダー:

```
PUT /DeviceManagerWebService/analytics/v1/services/agent-manager/actions/
refresh/invoke HTTP/1.1
Authorization: Basic c3lzdGVtOm1hbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/
3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22015
Accept: */*
Content-Type: application/json
Content-Length: 19
```

レスポンスヘッダー:

```
HTTP/1.1 200 OK
Date: Tue, 20 Oct 2015 07:07:36 GMT
Server: Cosminexus HTTP Server
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json;charset=UTF-8
```

レスポンスボディ:

```
{
  "state": "Succeeded"
}
```

3.9 HTM - Agents 情報取得時のパラメーターの取得

HTM - Agents 情報を取得する API のパラメータを取得します。

HTTP リクエスト構文 (URI)

```
GET <scheme>://<host>:<port>/<application>/<version>/services/agent-  
manager/actions/list
```

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

```
{  
  "name": "list",  
  "url": "<scheme>://<host>:<port>/<application>/<version>/services/  
agent-manager/actions/list/invoke",  
  "method": "PUT",  
  "type": "application/json",  
  "parameters": {  
    "agentType": "<エージェント種別>"  
  }  
}
```

実行例 (HTM - Agents 情報取得時のパラメーターの取得)

curl コマンドを使用したリクエストメッセージ:

```
curl -v -H "Content-Type: application/json" -u system:manager -X GET  
http://server_1:22015/DeviceManagerWebService/analytics/v1/services/  
agent-manager/actions/list
```

リクエストに対する実行結果

リクエストヘッダー:

```
GET /DeviceManagerWebService/analytics/v1/services/agent-manager/actions/  
list HTTP/1.1  
Authorization: Basic c3lzdGVtOmlhbmFnZXI=  
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/  
3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2  
Host: server_1:22015  
Accept: */*  
Content-Type: application/json
```

レスポンスヘッダー:

```
HTTP/1.1 200 OK  
Date: Tue, 20 Oct 2015 07:34:14 GMT  
Server: Cosminexus HTTP Server  
X-Frame-Options: SAMEORIGIN  
Transfer-Encoding: chunked  
Content-Type: application/json; charset=UTF-8
```

レスポンスボディ:

```
{  
  "name": "list",  
  "url": "http://server_1:22015/DeviceManagerWebService/analytics/v1/  
services/agent-manager/actions/list/invoke",  
  "method": "PUT",  
  "type": "application/json",  
  "parameters": {  
    "agentType": "all",  
    "formatVersion": "1.0"  
  }  
}
```

3.10 HTM - Agents 情報の取得

HTM - Agents 情報を取得する方法を示します。

HTTP リクエスト構文 (URI)

PUT <scheme>://<host>:<port>/<application>/<version>/services/agent-manager/actions/list/invoke

リクエストの構造

リクエストボディの構造を次に示します。

```
{
  "agentType" : "<エージェント種別>"
}
```

表 3-10 情報の取得時のリクエストボディのスキーマ

変数名	型	指定要否	説明
agentType	string	必須	エージェント種別 ・ HTM - Agent for RAID の場合 : raid ・ HTM - Agent for NAS の場合 : nas ・ すべてのエージェント種別の場合 : all

レスポンスの構造

```
{
  "data": [
    {
      "agentType": "...",
      "hostName": "...",
      "instanceName": "...",
      "model": "...",
      "serialNumber": "...",
      "smuIp": "...",
      "destIp": "...",
      "detourIp": "...",
      "lastMonitoredTime": "<YYYY-MM-DDThh:mm:ssZ>"
    },
    :
    :
  ]
}
```

表 3-11 情報の取得時のレスポンスボディのスキーマ

変数名	型	説明
data		
agentType	string	エージェントの種別 ・ HTM - Agent for RAID の場合 : RAID ・ HTM - Agent for NAS の場合 : NAS
hostName	string	エージェントのホスト名
instanceName	string	エージェントのインスタンス名
model	string	ストレージシステムの機種名※1
serialNumber	string	ストレージシステムのシリアル番号※1
smuIp	string	外部 NAS Manager を特定するための情報 外部 NAS Manager で管理される NAS Platform のインスタンスを作成するときに指定した SMU/NAS Manager IP Address が返却されます。

変数名	型	説明
destIp	string	Hitachi Virtual File Platform ^{※2} を特定するための情報 Hitachi Virtual File Platform ^{※2} のインスタンスを作成するときに指定した destination_address が返却されます。
detourIp	string	Hitachi Virtual File Platform ^{※2} を特定するための情報 Hitachi Virtual File Platform ^{※2} のインスタンスを作成するときに指定した detour_address が返却されます。
lastMonitoredTime	string	エージェントステータスを最後に更新した時間 YYYY-MM-DDThh:mm:ssZ

注※1

次の場合は null となります。

- Tuning Manager Agent REST API コンポーネントのサービス (Tuning Manager - Agent REST Web Service および Tuning Manager - Agent REST Application Service) が停止している、または、正常ではない状態で、HTM - Agents 情報の更新処理を実行した場合この場合は、次の手順を実施してください。

- a. 次のコマンドを実行して、Tuning Manager Agent REST API コンポーネントのサービスを停止します。

Windows の場合：

```
<エージェントのインストール先フォルダ>%htnm%bin%htmsrv stop -webservice
```

Linux の場合：

```
<エージェントのインストール先ディレクトリ>/htnm/bin/htmsrv stop -webservice
```

- b. htmRestDbEngineMessage#.log を確認して、エラーの要因を取り除きます。
htmRestDbEngineMessage#.log の格納先については、マニュアル「Hitachi Command Suite Tuning Manager - Agents」の Tuning Manager API 使用時のログ情報について説明している個所を参照してください。
- c. 次のコマンドを実行して、Tuning Manager Agent REST API コンポーネントのサービスを再起動します。

Windows の場合：

```
<エージェントのインストール先フォルダ>%htnm%bin%htmsrv start -webservice
```

Linux の場合：

```
<エージェントのインストール先ディレクトリ>/htnm/bin/htmsrv start -webservice
```

- d. HTM - Agents 情報の更新処理を再実行します。
- e. HTM - Agents 情報の取得処理を再実行します。

注※2

次のファイルサーバを総称して、「**Hitachi Virtual File Platform**」と呼びます。

- Hitachi Virtual File Platform
- Hitachi Capacity Optimization
- Hitachi Essential NAS Platform

実行例 (HTM - Agents 情報の取得)

curl コマンドを使用したリクエストメッセージ：

```
curl -v -u system:manager -H "Content-Type: application/json" -d '{"agentType":"all"}' -X PUT http://server_1:22015/
```

```
DeviceManagerWebService/analytics/v1/services/agent-manager/actions/list/  
invoke | python -m json.tool
```

リクエストに対する実行結果

リクエストヘッダー:

```
PUT /DeviceManagerWebService/analytics/v1/services/agent-manager/actions/  
list/invoke HTTP/1.1  
Authorization: Basic c3lzdGVtOmlhbmFnZXI=  
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/  
3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2  
Host: server_1:22015  
Accept: */*  
Content-Type: application/json  
Content-Length: 19
```

レスポンスヘッダー:

```
HTTP/1.1 200 OK  
Date: Tue, 20 Oct 2015 07:07:40 GMT  
Server: Cosminexus HTTP Server  
X-Frame-Options: SAMEORIGIN  
Transfer-Encoding: chunked  
Content-Type: application/json; charset=UTF-8
```

レスポンスボディ:

```
{  
  "data": [  
    {  
      "agentType": "RAID",  
      "hostName": "agent_1",  
      "instanceName": "VSPG1000_instance",  
      "model": "VSP G1000",  
      "serialNumber": "10182",  
      "lastMonitoredTime": "2015-10-20T07:07:37Z",  
      "formatVersion": "1.0"  
    },  
    {  
      "agentType": "RAID",  
      "hostName": "agent_1",  
      "instanceName": "VSP_instance",  
      "model": "VSP",  
      "serialNumber": "53039",  
      "lastMonitoredTime": "2015-10-20T07:07:37Z",  
      "formatVersion": "1.0"  
    },  
    {  
      "agentType": "RAID",  
      "hostName": "agent_1",  
      "instanceName": "USPV_instance",  
      "model": "USP V",  
      "serialNumber": "10007",  
      "lastMonitoredTime": "2015-10-20T07:07:37Z",  
      "formatVersion": "1.0"  
    }  
  ]  
}
```

3.11 アラート定義情報設定時のオブジェクト

アラート定義情報を設定するオブジェクトについて説明します。

3.11.1 アラート定義情報設定時の注意事項

アラート定義情報を設定する際の注意事項を次に示します。

- アラート定義数は、1 インスタンス当たり 500 個までです。
- email の送信先は、1 インスタンス当たり 8,192 個までです。
- リソースの絞り込み条件数は、1 インスタンス当たり 100,000 個までです。
- アラート定義情報で設定する値の大文字と小文字は区別されます。

なお、ここでは見やすくするために改行を入れていますが、実際に指定する際には改行を入れないでください。

3.11.2 アラート定義型

アラート定義型の構造を示します。

アラート定義情報の設定手順および定義例については、「[A.2 アラート定義情報の作成手順](#)」を参照してください。

```
{
  "alertDefName": "<アラート定義名>",
  "targets": [ <target 型: 監視対象> ],
  "notices": [ <notice 型: アクション> ],
  "triggers": [ <trigger 型: しきい値条件> ],
  "definitions": [ <definition 型: target, notice および trigger の組み合わせ> ]
}
```

表 3-12 アラート定義型のスキーマ

変数名	型	指定 要否	説明
alertDefName	string	必須	アラート定義名 アラート定義名は 127 バイト以内の次に示す文字で指定します。 半角英数字および#()·@_
targets	target 型の配 列	必須	アラート機能の監視対象のストレージシステムまたは NAS システム targets で指定できるストレージシステムまたは NAS システム数 は、5 個以内です。 詳細については、「 3.11.3 」を参照してください。
notices	notice 型の配 列	必須	アラート機能のアクションの定義 notices で指定できるアクションの定義数は、5 個以内です。 詳細については、「 3.11.4 」を参照してください。
triggers	trigger 型の 配列	必須	アラート機能の監視条件の定義 triggers で指定できる監視条件の定義数は、5 個以内です。 詳細については、「 3.11.5 」を参照してください。
definitions	definition 型 の配列	必須	target, notice および trigger の組み合わせ definitions で指定できる監視条件の定義数は、10 個以内です。 詳細については、「 3.11.6 」を参照してください。

3.11.3 target 型

監視対象を指定する型です。target 型の構造を示します。

```
{
  "targetID": "<target ID>",
  "targetDef": [ <targetDef 型: 監視対象の装置絞り込み条件> ]
}
```

表 3-13 target 型のスキーマ

変数名	型	指定要否	説明
targetID	string	必須	target ID target と notice と trigger の組み合わせを definitions で指定する場合に使用します。 target ID 名は 128 バイト以内の次に示す文字で指定します。 半角英数字および# () - @ _
targetDef	targetDef 型の配列	必須	アラート機能の監視対象のストレージシステムまたは NAS システムの特定

(1) targetDef 型

targetDef 型の構造を示します。

```
{
  "type": "<装置の特定>",
  "key": "<装置の特定条件>",
  "filters": "<filter 型: 装置の絞り込み条件>"
}
```

表 3-14 targetDef 型のスキーマ

変数名	型	指定要否	説明
type	string	必須	監視対象の装置の種別 指定できる値を次に示します。 <ul style="list-style-type: none"> model: ストレージシステムの機種名で対象装置を特定する場合に指定 block: ストレージシステムを、機種名およびシリアル番号で特定する場合に指定 vfp: Hitachi Virtual File Platform[※]を監視対象とする場合に指定 hnas: 外部 NAS Manager で管理される NAS Platform を監視対象とする場合に指定 unified: 内部 NAS Manager で管理される NAS Platform を監視対象とする場合に指定
key	model 情報型の配列 raid 情報型の配列 vfp 情報型の配列 smu 情報型の配列 unified 情報型の配列 のどれか	必須	監視対象の装置の特定条件 type に指定した値によって、特定条件の指定方法が異なります。 <ul style="list-style-type: none"> type に model を指定した場合: model 情報型 type に block を指定した場合: raid 情報型 type に vfp を指定した場合: vfp 情報型 type に hnas を指定した場合: smu 情報型 type に unified を指定した場合: unified 情報型
filters	filter 型の配列	任意	監視対象の装置のリソースの絞り込み条件 省略した場合は、全レコードのインスタンス (リソース) が対象となります。 filters 中の各要素は OR 条件となります。 filters で指定できる絞り込み条件数は、16,384 個以内です。

注※

次のファイルサーバを総称して、「Hitachi Virtual File Platform」と呼びます。

- Hitachi Virtual File Platform

- Hitachi Capacity Optimization
- Hitachi Essential NAS Platform

(2) model 情報型

model 情報型の構造を次に示します。

```
{
  "model" : "<ストレージシステムの機種名>"
}
```

表 3-15 model 情報型のスキーマ

変数名	型	指定要否	説明
model	string	必須	ストレージシステムの機種名※ HTM - Agent for RAID の PD レコードまたは PD_PDX レコードの DKC_NAME フィールドの値に相当します。 指定できる機種名の長さは、64 バイト以内です。

注※

ストレージシステムを VSP G1000 から VSP G1500 に上げた場合、アラート定義情報を削除した後、値を VSP G1000 から VSP G1500 に変更してアラート定義情報を登録し直す必要があります。

(3) raid 情報型

raid 情報型の構造を次に示します。

```
{
  "model" : "<ストレージシステムの機種名>",
  "serial" : "<シリアル番号>"
}
```

表 3-16 raid 情報型のスキーマ

変数名	型	指定要否	説明
model	string	必須	ストレージシステムの機種名※ HTM - Agent for RAID の PD レコードまたは PD_PDX レコードの DKC_NAME フィールドの値に相当します。 指定できる機種名の長さは、64 バイト以内です。
serial	string	必須	シリアル番号 HTM - Agent for RAID の PD レコードまたは PD_PDX レコードの DKC_SERIAL_NUMBER フィールドの値に相当します。 指定できるシリアル番号の長さは、32 バイト以内です。

注※

ストレージシステムを VSP G1000 から VSP G1500 に上げた場合、アラート定義情報を削除した後、値を VSP G1000 から VSP G1500 に変更してアラート定義情報を登録し直す必要があります。

(4) vfp 情報型

vfp 情報型の構造を次に示します。

```
{
```

```

    "nasSystemIp" : "<Hitachi Virtual File Platform を特定するための情報>"
  }

```

表 3-17 vfp 情報型のスキーマ

変数名	型	指定要否	説明
nasSystemIp	string	必須	Hitachi Virtual File Platform [※] を特定するための情報 Hitachi Virtual File Platform [※] のインスタンスを作成するときに指定した destination_address または detour_address を指定します。

注※

次のファイルサーバを総称して、「Hitachi Virtual File Platform」と呼びます。

- Hitachi Virtual File Platform
- Hitachi Capacity Optimization
- Hitachi Essential NAS Platform

(5) smu 情報型

smu 情報型の構造を次に示します。

```

{
  "smuIp" : "<外部 NAS Manager を特定するための情報 (SMU/NAS Manager IP Address) >"
}

```

表 3-18 smu 情報型のスキーマ

変数名	型	指定要否	説明
smuIp	string	必須	外部 NAS Manager を特定するための情報 外部 NAS Manager で管理される NAS Platform のインスタンスを作成するときに指定した SMU/NAS Manager IP Address を指定します。

(6) unified 情報型

unified 情報型の構造を次に示します。

```

{
  "model" : "<NAS モジュールを搭載したストレージシステムを特定するための情報 (機種名) >",
  "serial" : "<NAS モジュールを搭載したストレージシステムを特定するための情報 (シリアル番号) >"
}

```

表 3-19 unified 情報型のスキーマ

変数名	型	指定要否	説明
model	string	必須	NAS モジュールを搭載したストレージシステムの機種名 HTM・Agents 情報の取得の API で取得したストレージシステムの機種名を指定します。 指定できる機種名の長さは、64 バイト以内です。
serial	string	必須	NAS モジュールを搭載したストレージシステムのシリアル番号 インスタンスを作成するときに指定したストレージシステムのシリアル番号を指定します。 指定できるシリアル番号の長さは、32 バイト以内です。

(7) filter 型

filter 型の構造を次に示します。

```
{
  "filterConditions": [ <filterCondition 型 : 絞り込み条件> ]
}
```

表 3-20 filter 型のスキーマ

変数名	型	指定 要否	説明
filterConditions	filterCondition 型の配列	必須	監視対象の装置のリソースの絞り込み条件 絞り込みたいリソース定義の数だけ定義が必要です。 filterConditions を複数定義すると各 filterConditions は OR 条件となり、filterConditions の中の各要素は AND 条件となります。絞り込み条件の定義方法については、「(2)」を参照してください。

(8) filterCondition 型

filterCondition 型の構造を次に示します。

```
{
  "fieldName" : "<リソース定義名>" ,
  "value" : "<絞り込む値>"
}
```

表 3-21 filterCondition 型のスキーマ

変数名	型	指定 要否	説明
fieldName	string	必須	リソース定義名 指定できる文字列の長さは、41 バイト以内です。 リソース定義名の記載形式については、「(2)」を参照してください。
value	string	必須	絞り込む値 127 バイト以内の次に示す文字で指定します。 半角英数字、半角スペース、エスケープ文字*および! # & () * . / ; < = > @ _

注※

エスケープ文字については、「A.2.2 アラート定義情報を定義する際に使用できる文字について」を参照してください。

3.11.4 notice 型

アラート機能のアクションを指定する型です。notice 型の構造を示します。

```
{
  "noticeID": "<notice ID>" ,
  "emailNotification": "<notices を発生させる契機>" ,
  "emails": [ <email 型 : email アクション> ]
}
```

表 3-22 notice 型のスキーマ

変数名	型	指定 要否	説明
noticeID	string	必須	任意の notice ID target と notice と trigger の組み合わせを definitions で指定する場合に使用します。

変数名	型	指定要否	説明
			notice ID 名は 128 バイト以内の次に示す文字で指定します。 半角英数字および# () - @ _
emailNotification	string 型の配列	任意	email を送信する契機 warning, caution, または normal を指定できます。 複数選択できます。省略した場合、すべて指定しているとみなされます。推奨はすべての指定です。
emails	email 型の配列	任意	email アクションの設定 省略した場合は、定義なしとみなされます。 emails で指定できる email アクションの設定数は、3 個以内です。

(1) email 型

email 型の構造を示します。

```
{
  "addresses": "<email の宛先>",
  "subject": "<email のタイトル>",
  "body": "<email の本文>"
}
```

表 3-23 email 型のスキーマ

変数名	型	指定要否	説明
addresses	string の配列	必須	email の宛先 指定できる宛先の長さは 256 バイト以内、指定できる宛先の数は 10 個です。
subject	string	必須	email のタイトル タイトルは 300 バイト以内の次に示す文字で指定します。 半角英数字および! # & () * - . / ; < = > ? @ _
body	string	必須	email の本文 本文は 1,000 バイト以内の次に示す文字で指定します。 半角英数字、半角スペース、定義変数※、エスケープ文字※および! # & () * - . / ; < = > ? @ _

注※

定義変数およびエスケープ文字については「[A.2.2 アラート定義情報を定義する際に使用できる文字について](#)」を参照してください。

3.11.5 trigger 型

アラート機能の監視条件を指定する型です。trigger 型の構造を示します。

```
{
  "triggerID": "<trigger ID>",
  "isActive": "<設定したアラート定義情報の有効/無効>",
  "monitorAlways": "<アラート機能の状態>",
  "monitoringTime": [ <monitoringTime 型: 監視したい時間の範囲> ],
  "alertEveryTime": "<状態変化時の通知有無>",
  "evaluateAllData": "<全レコードの評価方法>",
  "evaluatePoR": "<値の存在を確認するかしないか>",
  "condition": [ <condition 型: しきい値の判定条件> ],
  "frequency": [ <frequency 型: 評価した値の通知頻度の条件> ]
}
```

表 3-24 trigger 型のスキーマ

変数名	型	指定要否	説明
triggerID	string	必須	任意の trigger ID target と notice と trigger の組み合わせを definitions で指定する場合に使用します。 trigger ID 名は 128 バイト以内の次に示す文字で指定します。 半角英数字および# () - @ _
isActive	string	任意	アラート定義情報の有効/無効 ・ 有効にする場合 : Y ・ 無効にする場合 : N 省略した場合は Y と見なされます。
monitorAlways	string	任意	アラート機能の状態 ・ 常に監視する場合 : Y ・ 常に監視しない場合 : N 省略した場合は Y と見なされます。
monitoringTime	monitoringTime 型	任意	監視したい時間の範囲 monitorAlways に N を指定した場合は、省略できません。
alertEveryTime	string	任意	状態変化時の通知有無 ・ 常に通知する場合 : Y ・ 常に通知しない場合 (状態が変化した場合だけ通知) : N 省略した場合は N と見なされます。
evaluateAllData	string	任意	全レコードの評価方法 ・ 全レコードをインスタンスごとに評価する場合 : Y ・ 全レコードをまとめて評価する場合 : N 省略した場合は Y と見なされます。
evaluatePoR	string	任意	条件として指定した値の存在の確認要否 ・ 値の存在を確認する場合 : Y ・ 値の存在を確認しない場合 : N 省略した場合は N と見なされます。
condition	condition 型	必須	しきい値の判定条件
frequency	frequency 型	任意	評価した値の通知頻度の条件

(1) monitoringTime 型

monitoringTime 型の構造を示します。

```
{
  "starttime": "<開始時間>",
  "endtime": "<終了時間>"
}
```

表 3-25 monitoringTime 型のスキーマ

変数名	型	指定要否	説明
starttime	string	必須	開始時間 hh:mm (UTC) の形式で指定します。 指定できる値を次に示します。 ・ hh : 00~23 ・ mm : 00~59
endtime	string	必須	終了時間 hh:mm (UTC) の形式で指定します。 指定できる値を次に示します。 ・ hh : 00~23 ・ mm : 00~59

(2) condition 型

condition 型の構造を示します。

```
{
  "targetOfPoR": "<存在を確認する値>",
  "warning": "<異常値の条件式>",
  "caution": "<警告値の条件式>"
}
```

表 3-26 condition 型のスキーマ

変数名	型	指定要否	説明
targetOfPoR	string	任意	存在を確認する値 条件式で指定します。条件式の指定方法については、「A.2.1」の「(4)」を参照してください。 trigger 型の evaluatePoR が Y の場合、省略できません。 省略した場合は定義なしとします。 指定できる条件式の長さは、200 バイト以内です。
warning	string	任意	異常値の条件式 条件式の指定方法については、「A.2.1」の「(4)」を参照してください。 trigger 型の evaluatePoR が N の場合、省略できません。 省略した場合は定義なしとします。 指定できる条件式の長さは、2,500 バイト以内です。 また、指定できる条件式の数、10 個以内です。
caution	string	任意	警告値の条件式 条件式の指定方法については、「A.2.1」の「(4)」を参照してください。 trigger 型の evaluatePoR が N の場合、省略できません。 省略した場合は定義なしとします。 指定できる条件式の長さは、2,500 バイト以内です。 また、指定できる条件式の数、10 個以内です。

(3) frequency 型

frequency 型の構造を示します。

```
{
  "count": "n / m"
}
```

表 3-27 frequency 型のスキーマ

変数名	型	指定要否	説明
count	string	必須	しきい値を超えた回数(n)／インターバル数(m) 何回中何回しきい値を超えたらイベントを発行するかを指定します。 指定できる値を次に示します。 <ul style="list-style-type: none"> n : 1 以上 m 以下の整数 m : n 以上 30 以下の整数

表 3-28 発生頻度と通知のタイミング

発生頻度	通知のタイミング
$n < m$	m インターバル中、 n 回しきい値を超えた場合にアラートイベントが通知されます。以降 m 回アラートを評価する間に n 回しきい値を超えるごとにアラートイベントが通知されます。
$n = m$	しきい値を一度超えるとアラートイベントが通知され、以降しきい値を超え続けている間、 n 回ごとにアラートイベントが通知されます。

3.11.6 definition 型

target, notice および trigger の組み合わせを指定する型です。definition 型の構造を示します。

```
{
  "targetIDs": [ <targetID のリスト> ],
  "noticeIDs": [ <noticeID のリスト> ],
  "triggerIDs": [ <triggerID のリスト> ]
}
```

表 3-29 definition 型のスキーマ

変数名	型	指定要否	説明
targetIDs	string 型の配列	必須	targetID のリスト 指定できる targetID の数は、5 個以内です。
noticeIDs	string 型の配列	必須	noticeID のリスト 指定できる noticeID の数は、5 個以内です。
triggerIDs	string 型の配列	必須	triggerID のリスト 指定できる triggerID の数は、5 個以内です。

注意

リソースの絞り込み条件でレコードを指定している target 型の場合、同じレコードを使用して定義している trigger 型としか組み合わせることはできません。

パフォーマンスデータを取得するための API

この章では、パフォーマンスデータを取得する前に実行が必要な API、およびパフォーマンスデータを取得する API について、説明します。

なお、実行例には見やすくするために改行を入れています。

- [4.1 Tuning Manager API のリソース一覧](#)
- [4.2 HTM - Agents 一覧の手動リフレッシュ処理実行時のパラメーターの取得](#)
- [4.3 HTM - Agents 一覧の手動リフレッシュ処理の実行](#)
- [4.4 HTM - Agents のインスタンス一覧の取得](#)
- [4.5 HTM - Agent for RAID のインスタンス一覧の取得](#)
- [4.6 パフォーマンスデータの取得](#)

4.1 Tuning Manager API のリソース一覧

Tuning Manager API のリソース一覧を次の表に示します。なお、リソースは URI で表記していません。

表 4-1 Tuning Manager API のリソース一覧

API の用途	リクエスト		参照先	
	HTTP method	URI		
パフォーマンスデータを取得する前に実行が必要な API				
HTM - Agents 一覧の手動リフレッシュ実行時のパラメーターの取得	GET	/<application>/v1/services/AgentManager/actions/refresh	4.2	
HTM - Agents 一覧の手動リフレッシュ処理の実行	POST	/<application>/v1/services/AgentManager/actions/refresh/invoke	4.3	
HTM - Agents のインスタンス一覧の取得	GET	/<application>/v1/objects/Agents	4.4	
HTM - Agent for RAID のインスタンス一覧の取得	GET	/<application>/v1/objects/AgentForRAID	4.5	
パフォーマンスデータを取得する API (HTM - Agent for RAID) ※1				
論理デバイスのパフォーマンスデータの取得	PD_ELC レコード	GET	/<application>/v1/objects/RAID_PD_ELC	4.6
	PD_LDC レコード	GET	/<application>/v1/objects/RAID_PD_LDC	4.6
	PD_LSEC レコード	GET	/<application>/v1/objects/RAID_PD_LSEC	4.6
	PI_LDE, PI_LDE1, PI_LDE2 および PI_LDE3 レコード	GET	/<application>/v1/objects/RAID_PI_LDE	4.6
	PI_LDS, PI_LDS1, PI_LDS2 および PI_LDS3 レコード	GET	/<application>/v1/objects/RAID_PI_LDS	4.6
ポートのパフォーマンスデータの取得	PD_PTC レコード	GET	/<application>/v1/objects/RAID_PD_PTC	4.6
	PI_PTS レコード	GET	/<application>/v1/objects/RAID_PI_PTS	4.6
パリティグループのパフォーマンスデータの取得	PD_RGC レコード	GET	/<application>/v1/objects/RAID_PD_RGC	4.6
	PI_RGS レコード	GET	/<application>/v1/objects/RAID_PI_RGS	4.6
ストレージシステムのパフォーマンスデータの取得	PD レコード	GET	/<application>/v1/objects/RAID_PD	4.6
	PI レコード	GET	/<application>/v1/objects/RAID_PI	4.6
	PI_LDA レコード	GET	/<application>/v1/objects/RAID_PI_LDA	4.6
CLPR のパフォーマンスデータの取得	PD_CLPC レコード	GET	/<application>/v1/objects/RAID_PD_CLPC	4.6

APIの用途		リクエスト		参照先
		HTTP method	URI	
	PI_CLCS レコード	GET	/<application>/v1/objects/RAID_PI_CLCS	4.6
	PI_CLPS レコード	GET	/<application>/v1/objects/RAID_PI_CLPS	4.6
MP Blade ごとの CLPR のパフォーマンスデータの取得	PI_CLMS レコード	GET	/<application>/v1/objects/RAID_PI_CLMS	4.6
プロセッサのパフォーマンスデータの取得	PD_UMS レコード	GET	/<application>/v1/objects/RAID_PD_UMS	4.6
	PI_PRCs レコード	GET	/<application>/v1/objects/RAID_PI_PRCs	4.6
CHA のパフォーマンスデータの取得	PI_CHS レコード	GET	/<application>/v1/objects/RAID_PI_CHS	4.6
ジャーナルグループのパフォーマンスデータの取得	PI_JNLS レコード	GET	/<application>/v1/objects/RAID_PI_JNLS	4.6
ドライブのパフォーマンスデータの取得	PI_PDOS レコード	GET	/<application>/v1/objects/RAID_PI_PDOS	4.6
プールのパフォーマンスデータの取得	PD_PLC レコード	GET	/<application>/v1/objects/RAID_PD_PLC	4.6
	PD_PLF レコード	GET	/<application>/v1/objects/RAID_PD_PLF	4.6
	PD_PLR レコード	GET	/<application>/v1/objects/RAID_PD_PLR	4.6
	PI_PLS レコード	GET	/<application>/v1/objects/RAID_PI_PLS	4.6
V-VOL のパフォーマンスデータの取得	PD_VVC レコード	GET	/<application>/v1/objects/RAID_PD_VVC	4.6
	PD_VVF レコード	GET	/<application>/v1/objects/RAID_PD_VVF	4.6
Tier (Pool) のパフォーマンスデータの取得	PD_PLTC レコード	GET	/<application>/v1/objects/RAID_PD_PLTC	4.6
	PD_PLTR レコード	GET	/<application>/v1/objects/RAID_PD_PLTR	4.6
	PD_PLTS レコード	GET	/<application>/v1/objects/RAID_PD_PLTS	4.6
	PI_PLTI レコード	GET	/<application>/v1/objects/RAID_PI_PLTI	4.6
Tier (V-VOL) のパフォーマンスデータの取得	PD_VVTC レコード	GET	/<application>/v1/objects/RAID_PD_VVTC	4.6
	PI_VVTI レコード	GET	/<application>/v1/objects/RAID_PI_VVTI	4.6
容量拡張機能, 容量削減機能, スナップショット, およびプロビジョニングの合計効果 (システムデータを	PD_SEFF レコード	GET	/<application>/v1/objects/RAID_PD_SEFF	4.6
	PD_PEFF レコード	GET	/<application>/v1/objects/RAID_PD_PEFF	4.6

API の用途		リクエスト		参照先
		HTTP method	URI	
除く) のパフォーマンスデータの取得				
パフォーマンスデータを取得する API (HTM - Agent for RAID (TCP/IP 接続を使用しての収集)) ※1, ※2				
論理デバイスのパフォーマンスデータの取得	PI_ELDS レコード	GET	/<application>/v1/objects/RAID_PI_ELDS	4.6
	PI_LSP レコード	GET	/<application>/v1/objects/RAID_PI_LSP	4.6
	PI_LDSX レコード	GET	/<application>/v1/objects/RAID_PI_LDSX	4.6
	PI_LDTC レコード	GET	/<application>/v1/objects/RAID_PI_LDTC	4.6
	PI_LDUR レコード	GET	/<application>/v1/objects/RAID_PI_LDUR	4.6
	PI_LTSP レコード	GET	/<application>/v1/objects/RAID_PI_LTSP	4.6
	PI_LDU レコード	GET	/<application>/v1/objects/RAID_PI_LDU	4.6
	PI_LUSP レコード	GET	/<application>/v1/objects/RAID_PI_LUSP	4.6
ポートのパフォーマンスデータの取得	PI_MPTS レコード	GET	/<application>/v1/objects/RAID_PI_MPTS	4.6
	PI_PTSX レコード	GET	/<application>/v1/objects/RAID_PI_PTSX	4.6
アクセスパスのパフォーマンスデータの取得	PI_CBMB レコード	GET	/<application>/v1/objects/RAID_PI_CBMB	4.6
	PI_CCMS レコード	GET	/<application>/v1/objects/RAID_PI_CCMS	4.6
	PI_CHAC レコード	GET	/<application>/v1/objects/RAID_PI_CHAC	4.6
	PI_CPS レコード	GET	/<application>/v1/objects/RAID_PI_CPS	4.6
	PI_DBMB レコード	GET	/<application>/v1/objects/RAID_PI_DBMB	4.6
	PI_DKAC レコード	GET	/<application>/v1/objects/RAID_PI_DKAC	4.6
	PI_MBDR レコード	GET	/<application>/v1/objects/RAID_PI_MBDR	4.6
	PI_MPCS レコード	GET	/<application>/v1/objects/RAID_PI_MPCS	4.6
パリティグループのパフォーマンスデータの取得	PI_EVGS レコード	GET	/<application>/v1/objects/RAID_PI_EVGS	4.6
	PI_RGSX レコード	GET	/<application>/v1/objects/RAID_PI_RGSX	4.6
	PI_RGU レコード	GET	/<application>/v1/objects/RAID_PI_RGU	4.6

APIの用途		リクエスト		参照先
		HTTP method	URI	
ストレージシステムのパフォーマンスデータの取得	PD_PDX レコード	GET	/<application>/v1/objects/RAID_PD_PDX	4.6
	PI_SCPS レコード	GET	/<application>/v1/objects/RAID_PI_SCPS	4.6
	PI_SCS レコード	GET	/<application>/v1/objects/RAID_PI_SCS	4.6
MP Blade ごとの CLPR のパフォーマンスデータの取得	PI_CLMX レコード	GET	/<application>/v1/objects/RAID_PI_CLMX	4.6
プロセッサのパフォーマンスデータの取得	PD_UMSX レコード	GET	/<application>/v1/objects/RAID_PD_UMSX	4.6
	PI_MPS レコード	GET	/<application>/v1/objects/RAID_PI_MPS	4.6
CHA のパフォーマンスデータの取得	PI_CHDR レコード	GET	/<application>/v1/objects/RAID_PI_CHDR	4.6
ディスクアダプターのパフォーマンスデータの取得	PI_DKDR レコード	GET	/<application>/v1/objects/RAID_PI_DKDR	4.6
	PI_HWSP レコード	GET	/<application>/v1/objects/RAID_PI_HWSP	4.6
	PI_HBAS レコード	GET	/<application>/v1/objects/RAID_PI_HBAS	4.6
ジャーナルグループのパフォーマンスデータの取得	PI_JNLX レコード	GET	/<application>/v1/objects/RAID_PI_JNLX	4.6
パフォーマンスデータを取得する API (NAS Platform を監視している HTM・Agent for NAS) ※1				
SMU/NAS Manager のパフォーマンスデータの取得	PD_HSMU レコード	GET	/<application>/v1/objects/NAS_PD_HSMU	4.6
NAS Platform のノードごとのパフォーマンスデータの取得	PD_HNC レコード	GET	/<application>/v1/objects/NAS_PD_HNC	4.6
	PI_HNCS レコード	GET	/<application>/v1/objects/NAS_PI_HNCS	4.6
	PI_HNHS レコード	GET	/<application>/v1/objects/NAS_PI_HNHS	4.6
	PI_HNS レコード	GET	/<application>/v1/objects/NAS_PI_HNS	4.6
EVS のパフォーマンスデータの取得	PD_HEC レコード	GET	/<application>/v1/objects/NAS_PD_HEC	4.6
ファイルシステムのパフォーマンスデータの取得	PD_HFSC レコード	GET	/<application>/v1/objects/NAS_PD_HFSC	4.6
	PI_HFSS レコード	GET	/<application>/v1/objects/NAS_PI_HFSS	4.6
ストレージプールのパフォーマンスデータの取得	PD_HPLC レコード	GET	/<application>/v1/objects/NAS_PD_HPLC	4.6
システムドライブごとのパフォーマンスデータの取得	PD_HSDC レコード	GET	/<application>/v1/objects/NAS_PD_HSDC	4.6

API の用途	リクエスト		参照先
	HTTP method	URI	
パフォーマンスデータを取得する API (Hitachi Virtual File Platform ^{※3} を監視している HTM - Agent for NAS) ^{※1}			
プロセスのパフォーマンスデータの取得	PD レコード	GET	<code><application>/v1/objects/NAS_PD</code> 4.6
NAS システムのパフォーマンスデータの取得	PD_CHC レコード	GET	<code><application>/v1/objects/NAS_PD_CHC</code> 4.6
	PD_CPC レコード	GET	<code><application>/v1/objects/NAS_PD_CPC</code> 4.6
	PD_IAC レコード	GET	<code><application>/v1/objects/NAS_PD_IAC</code> 4.6
	PI レコード	GET	<code><application>/v1/objects/NAS_PI</code> 4.6
ローカルディスクデバイスのパフォーマンスデータの取得	PI_DEVD レコード	GET	<code><application>/v1/objects/NAS_PI_DEVD</code> 4.6
	PI_DEVS レコード	GET	<code><application>/v1/objects/NAS_PI_DEVS</code> 4.6
ファイルシステムのパフォーマンスデータの取得	PD_FSC レコード	GET	<code><application>/v1/objects/NAS_PD_FSC</code> 4.6
	PD_FSL レコード	GET	<code><application>/v1/objects/NAS_PD_FSL</code> 4.6

注※1

objects ドメインの下の<object-type>に指定したオブジェクトの型に相当するレコード情報を出力します。

注※2

監視対象ストレージシステムが VSP Gx00 モデル, VSP Fx00 モデル, VSP E990, HUS VM, VSP 5000 シリーズ, VSP G1000, G1500, VSP F1500, および Virtual Storage Platform シリーズの場合で, HTM - Agent for RAID のインストールとセットアップ時に, TCP/IP 接続を使用して収集できるパフォーマンスデータを取得するための設定をしている場合にだけ取得できるパフォーマンスデータです。TCP/IP 接続を使用して収集できるパフォーマンスデータを取得するための設定や前提条件についての詳細は, マニュアル「Hitachi Command Suite Tuning Manager - Agents」の付録の TCP/IP 接続を使用して収集できるパフォーマンスデータについて説明している章を参照してください。

また, これらのパフォーマンスデータの情報を収集する際, 指定できるクエリのキー値には制限があります。詳細については「4.6 パフォーマンスデータの取得」を参照してください。

注※3

次のファイルサーバを総称して, 「Hitachi Virtual File Platform」と呼びます。

- Hitachi Virtual File Platform
- Hitachi Capacity Optimization
- Hitachi Essential NAS Platform

これらの API の具体的な使用方法は, 「付録 A. Tuning Manager API の使用方法」を参照してください。

4.2 HTM - Agents 一覧の手動リフレッシュ処理実行時のパラメーターの取得

HTM - Agents の一覧の手動リフレッシュ処理実行時に必要なパラメーターの取得方法について示します。

この API は、Tuning Manager API が提供する 1 つのアクションに関して、parameters フィールドにデフォルト値が指定された Action オブジェクトを API クライアントに返します。この Action オブジェクトはそのまま「4.3 HTM - Agents 一覧の手動リフレッシュ処理の実行」で示す API の引数として使用できます。

HTTP リクエスト構文 (URI)

```
GET <scheme>://<host>:<port>/<application>/<version>/services/AgentManager/actions/refresh
```

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

レスポンスボディの構造を次に示します。

```
{
  "name": "RefreshAgentInstancesParameters",
  "url": "<scheme>://<host>:<port>/<application>/<version>/services/AgentManager/actions/refresh/invoke",
  "method": "POST",
  "type": "application/json",
  "parameters": [{"type": "string", "value": "All"}]
}
```

実行例 (HTM - Agents 一覧の手動リフレッシュ処理実行時のパラメーターの取得)

curl コマンドを使用したリクエストメッセージ:

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

リクエストに対する実行結果

リクエストヘッダー:

```
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
```

レスポンスヘッダー:

```
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
```

レスポンスボディ:

```

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

```

4.3 HTM - Agents 一覧の手動リフレッシュ処理の実行

HTM - Agents の一覧情報の手動リフレッシュは、HTM - Agents で監視対象システムやインスタンス情報を変更した場合など構成変更したときには、パフォーマンスデータを取得する前に実行する必要があります。

HTM - Agents の一覧情報を手動でリフレッシュする方法について示します。

HTTP リクエスト構文 (URI)

POST <scheme>://<host>:<port>/<application>/<version>/services/AgentManager/actions/refresh/invoke

リクエストの構造

リクエストボディの構造を次に示します。

```

{
  "name"      : "RefreshAgentInstancesParameters",
  "url"       : "<scheme>://<host>:<port>/<application>/<version>/
services/AgentManager/actions/refresh/invoke",
  "method"    : "POST",
  "type"      : "application/json",
  "parameters": [ { ... } ]
}

```

parameters に指定できる値について、次の表に示します。

表 4-2 HTM - Agents 一覧の手動リフレッシュ時の parameters

変数名	値		要素数	説明
parameters	type	string : 変数の型を指します。	1	更新する対象の Agent 種別を定義します。 要素が 2 つ以上指定した場合、第 2 要素以降は無視されます。
	value	次の値のどれかでリフレッシュ対象を指定します。 <ul style="list-style-type: none"> All : Tuning Manager API での情報取得をサポートするすべてのエージェントを指します。 AgentForRAID : HTM - Agent for RAID を指します。 AgentForNAS : HTM - Agent for NAS を指します。 		

レスポンスの構造

レスポンスボディの構造を次に示します。

```

{"status": "Completed"}

```

正常に更新処理が完了した場合は、レスポンスボディには次の表に示す Job オブジェクトを含みます。エラーが発生した場合は、発生した原因によって Job オブジェクトが含まれる場合と、Error

オブジェクトが含まれる場合があります。レスポンスボディがどちらのオブジェクトであるかはステータスコードで判別できます。

表 4-3 Job オブジェクトのスキーマ

変数名	型	説明
status	string	処理結果の状態。 処理が正常に完了した場合は「Completed」、エラーが発生した場合は「Error」とします。
errorID	string	発生したエラーを一意に識別できる識別子。 エラー発生時はログ出力のエラー ID と同じ値になります。なお、正常時にはこの変数は付与されません。
errorDescription	string	発生したエラーの説明。 エラー発生時はログ出力と同じ文字列になります。また、文字列は英語で出力されます。なお、正常時にはこの変数は付与されません。

HTM - Agents 一覧の手動リフレッシュ処理の実行で発生するエラー時のレスポンスについて次の表に示します。

表 4-4 HTM - Agents 一覧の手動リフレッシュ時のエラーレスポンス

エラー発生条件	HTTP レスポンス			
	ステータスコード	Body		
		Error オブジェクト	Job オブジェクト	
		messageID	Status	errorID
PFM - Manager が停止しています	200	—	"Error"	KATR11002-E
Action オブジェクトの parameters の第 1 要素にサポート外の Agent 種別が指定されました	400	KATR11032-E	—	—

(凡例)

— : 該当なし

実行例

- ・ 実行例 1 (すべてのエージェントの一覧をリフレッシュする場合)
- ・ 実行例 2 (HTM - Agent for RAID の一覧をリフレッシュする場合)

実行例 1 (すべてのエージェントの一覧をリフレッシュする場合)

curl コマンドを使用したリクエストメッセージ :

```
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" } ] }'
```

リクエストに対する実行結果 /

リクエストヘッダー :

```
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
```

レスポンスヘッダー :

```
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
```

レスポンスボディ :

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

実行例 2 (HTM - Agent for RAID の一覧をリフレッシュする場合)

curl コマンドを使用したリクエストメッセージ :

```
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" } ] }'
```

リクエストに対する実行結果

リクエストヘッダー :

```
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
```

レスポンスヘッダー :

```
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
```

レスポンスボディ :

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

4.4 HTM - Agents のインスタンス一覧の取得

HTM - Agents のインスタンスの一覧を取得する方法について示します。

HTTP リクエスト構文 (URI)

```
GET <scheme>://<host>:<port>/<application>/<version>/objects/Agents?<query-string>
```

クエリ文字列

クエリ文字の指定例：

```
agentType=RAID
```

表 4-5 HTM - Agents のインスタンス一覧取得時に指定可能なクエリ文字列のキーと値

キー	値		指定要否
	値域	概要	
agentType	指定できる値を次に示します。 <ul style="list-style-type: none">ALL : Tuning Manager API での情報取得をサポートするすべてのエージェントを指します。RAID : HTM - Agent for RAID を指します。NAS : HTM - Agent for NAS を指します。	インスタンス一覧を取得したい HTM - Agents の Agent 種別	必須

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

レスポンスボディの構造を次に示します。

- agentType が RAID の場合

```
{
  "items": [{
    "agentType": "RAID",
    "instanceName": "...",
    "hostName": "...",
    "ipAddr": "...",
    "dataModelVersion": "...",
    "storageVendor": "...",
    "storageModel": "...",
    "storageSerialNumber": "...",
    "lastUpdateTime": "...",
  },
  :
  :
  ]
}
```

- agentType が NAS かつ nasMonitoringType が Unified NAS の場合

```
{
  "items": [{
    "agentType": "NAS",
    "instanceName": "...",
    "hostName": "...",
    "ipAddr": "...",
    "dataModelVersion": "...",
    "nasMonitoringType": "Unified NAS",
  }
]
```

```

"svpIpAddr": "...",
"storageModel": "...",
"storageSerialNumber": "...",
"nasNodes": [{
  "nodeName": "...",
  "clusterUuid": "...",
  },
  :
  :
]
"nasClusters": [{
  "clusterName": "...",
  "clusterUuid": "...",
  },
  :
  :
]
"lastUpdateTime": "...",
},
:
:
]
}

```

- agentType が NAS かつ nasMonitoringType が External SMU の場合

```

{
  "items": [{
    "agentType": "NAS",
    "instanceName": "...",
    "hostName": "...",
    "ipAddr": "...",
    "dataModelVersion": "...",
    "nasMonitoringType": "External SMU",
    "smuIpAddr": "...",
    "nasNodes": [{
      "nodeName": "...",
      "nodeUuid": "...",
      "clusterUuid": "...",
      },
      :
      :
    ]
    "nasClusters": [{
      "clusterName": "...",
      "clusterUuid": "...",
      },
      :
      :
    ]
    "lastUpdateTime": "...",
  },
  :
  :
]
}

```

- agentType が NAS かつ nasMonitoringType が VFP の場合

```

{
  "items": [{
    "agentType": "NAS",
    "instanceName": "...",
    "hostName": "...",
    "ipAddr": "...",
    "dataModelVersion": "...",
    "nasMonitoringType": "VFP",
    "destIpAddr": "...",
    "detourIpAddr": "...",
    "lastUpdateTime": "...",
  },
  :
  :
]
}

```

レスポンスボディに含むデータフォーマットを次の表に示します。

表 4-6 HTM - Agents のインスタンス一覧取得時のデータフォーマット

オブジェクト名	個数	説明
Collection	1	Agents オブジェクトを子に持つ Collection オブジェクト
- Agents	-	AgentForRAID オブジェクトおよび AgentForNAS オブジェクトの総称
- AgentForRAID	0..n	AgentForRAID オブジェクト スキーマの詳細については「表 4-7」に示します。
- AgentForNAS	0..n	AgentForNAS オブジェクト スキーマの詳細については「表 4-7」に示します。

表 4-7 Agents オブジェクトのスキーマ

変数名	agentType および nasMonitoringType ごとの表示の有無				型	説明
	RAID	NAS				
		Unified NAS	External SMU	VFP		
agentType	○	○	○	○	string	HTM - Agents の Agent 種別
instanceName	○	○	○	○	string	HTM - Agents のインスタンス名。
hostName	○	○	○	○	string	HTM - Agents が稼働しているホストの監視ホスト名。
ipAddr	○	○	○	○	string	HTM - Agents が稼働しているホストの IP アドレス。
dataModelVersion	○	○	○	○	decimal	HTM - Agents のデータモデルバージョン。
nasMonitoringType	×	○※1	○※1	○※1	string	NAS システムの種別。 <ul style="list-style-type: none"> 内部 NAS Manager で管理される NAS Platform の場合：Unified NAS 外部 NAS Manager で管理される NAS Platform の場合：External SMU Hitachi Virtual File Platform の場合：VFP
destIpAddr	×	×	×	○※1	string	NAS システムの IP アドレス。 なお、nasMonitoringType が null の場合、この要素は表示されません。
detourIpAddr	×	×	×	○※1	string	NAS システムの予備の IP アドレス。 なお、nasMonitoringType が null の場合、この要素は表示されません。
smuIpAddr	×	×	○※2	×	string	外部 NAS Manager の IP アドレス。

変数名	agentType および nasMonitoringType ごとの表示の有 無				型	説明
	RAID	NAS				
		Unified NAS	External SMU	VFP		
						なお、nasMonitoringType が null の場合、この要素は表示されません。
svpIpAddr	×	○※1	×	×	string	NAS モジュールを搭載したストレージシステムを管理している SVP の IP アドレス。 なお、nasMonitoringType が null の場合、この要素は表示されません。
storageVendor	○ ※3	×	×	×	string	ストレージシステム※4 のベンダ名。
storageModel	○ ※3	○※1	×	×	string	ストレージシステム※4 のモデル名。
storageSerialNumber	○ ※3	○※1	×	×	string	ストレージシステム※4 の製造番号。
nasNodes	×	○※5	○※5	×	配列	nasNodes 属性のスキーマについては、「表 4-8」に示します。 なお、nasMonitoringType が null の場合、この要素は表示されません。
nasClusters	×	○※5	○※5	×	配列	nasClusters 属性のスキーマについては、「表 4-9」に示します。 なお、nasMonitoringType が null の場合、この要素は表示されません。
lastUpdateTime	○ ※3	○※5	○※6	○ ※7	string	HTM - Agents の情報の更新処理が成功した時刻。 ISO8601 の拡張形式 (YYYY-MM-DDThh:mm:ssZ) で表します。hh は 24 時間表記です。また、Z は UTC 時刻であることを表します。

(凡例)

- : 表示される
- × : 表示されない

注※1

次の場合は null となります。

- インスタンスの設定に不正がある場合
- インスタンスが起動していない状態でエージェント情報をリフレッシュした場合

注※2

次の場合は null となります。

- インスタンスの設定に不正がある場合
- PD_HSMU レコードが収集されていない、または Tuning Manager Agent REST API コンポーネントのサービス (Tuning Manager - Agent REST Web Service および Tuning

Manager - Agent REST Application Service) が起動していない状態でエージェント情報をリフレッシュした場合

注※3

次の場合は null となります。

- 監視対象ストレージシステムが存在しない場合
- インスタンスの設定に不正がある場合
- PD レコードまたは PD_PDX レコードが収集されていない状態でエージェント情報をリフレッシュした場合
- 監視対象ストレージシステムが VSP Gx00 モデル、VSP Fx00 モデル、VSP E990、HUS VM、VSP 5000 シリーズ、VSP G1000、G1500、VSP F1500、Virtual Storage Platform シリーズのときに、パフォーマンスデータの収集にコマンドデバイスを使用していない場合

注※4

監視対象ストレージシステムには仮想ストレージマシンは含まれません。

注※5

次の場合は null となります。

- インスタンスの設定に不正がある場合
- PD_HNC レコードが収集されていない、または Tuning Manager Agent REST API コンポーネントのサービス (Tuning Manager - Agent REST Web Service および Tuning Manager - Agent REST Application Service) が起動していない状態でエージェント情報をリフレッシュした場合

注※6

次の場合は null となります。

- インスタンスの設定に不正がある場合
- PD_HSMU レコードおよび PD_HNC レコードが収集されていない、または Tuning Manager Agent REST API コンポーネントのサービス (Tuning Manager - Agent REST Web Service および Tuning Manager - Agent REST Application Service) が起動していない状態でエージェント情報をリフレッシュした場合

注※7

次の場合は null となります。

- インスタンスの設定に不正がある場合
- インスタンスが起動していない、PD_CPC レコードが収集されていない、または Tuning Manager Agent REST API コンポーネントのサービス (Tuning Manager - Agent REST Web Service および Tuning Manager - Agent REST Application Service) が起動していない状態でエージェント情報をリフレッシュした場合

表 4-8 nasNodes 属性のスキーマ

変数名	型	説明
nodeName	string	監視対象ノードのノード名。
nodeUuid	string	監視対象ノードの UUID。 非クラスタノードの場合だけ表示されます。
clusterUuid	string	監視対象ノードが属するクラスタの UUID。 クラスタ構成のノードの場合だけ表示されます。

表 4-9 nasClusters 属性のスキーマ

変数名	型	説明
clusterName	string	監視対象クラスタのクラスタ名。
clusterUuid	string	監視対象クラスタの UUID。

HTM - Agents のインスタンス一覧取得で発生するエラー時のレスポンスについて次の表に示します。

表 4-10 HTM - Agents のインスタンス一覧取得時のエラーレスポンス

エラー発生条件	HTTP レスポンス	
	ステータスコード	Body に含まれるエラー ID
「agentType」をクエリで指定していません。	400	KATR11037-E
「agentType=」の値に不正な Agent 種別を指定しています。	400	KATR11038-E
内部エラーが発生しました。	500	KATR00101-E

実行例 (HTM - Agents のインスタンス一覧の取得)

curl コマンドを使用したリクエストメッセージ:

```
curl -v -u system:manager -H Content-Type: application/json -X GET
http://server_1:22015/TuningManager/v1/objects/Agents?agentType=ALL
```

リクエストに対する実行結果

リクエストヘッダー:

```
GET /TuningManager/v1/objects/Agents?agentType=ALL HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7
NSS/3.18 Basic ECC zlib/1.2.3 libidn/1.18 libssh2/1.4.2
Host: server_1:22015
Host: :22016
Accept: */*
Content-Type: application/json
```

レスポンスヘッダー:

```
HTTP/1.1 200 OK
Date: Mon, 28 Sep 2015 06:41:14 GMT
Server: Cosminexus HTTP Server
Last-Modified: Mon, 28 Sep 2015 06:41:03 GMT
Cache-Control: no-cache
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json; charset=utf-8
```

レスポンスボディ:

```
{
  "items": [{
    "agentType": "RAID",
    "instanceName": "MONITOR1",
    "hostName": "esxi4VM02",
    "ipAddr": "<MONITOR1 の IP アドレス>",
    "dataModelVersion": "9.0",
    "storageVendor": "HITACHI",
    "storageModel": "HUS110",
    "storageSerialNumber": "91210942",
    "lastUpdateTime": "2015-09-09T04:47:16Z"
  }],
  {
```

```

    "agentType": "NAS",
    "instanceName": "MONITOR1",
    "hostName": "esxi4VM02",
    "ipAddr": "<MONITOR1 の IP アドレス>",
    "dataModelVersion": 6.6,
    "nasMonitoringType": VFP,
    "destIpAddr": "<監視対象 NAS システムの IP アドレス>",
    "detourIpAddr": "",
    "lastUpdateTime": "2015-09-09T04:47:16Z"
  }
}

```

4.5 HTM - Agent for RAID のインスタンス一覧の取得

HTM - Agent for RAID のインスタンスの一覧を取得する方法について示します。

HTTP リクエスト構文 (URI)

```
GET <scheme>://<host>:<port>/<application>/<version>/objects/AgentForRAID
```

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

レスポンスボディの構造を次に示します。

```

{
  "items": [
    {
      "instanceName": "...",
      "hostName": "...",
      "ipAddr": "...",
      "dataModelVersion": "...",
      "lastUpdateTime": "...",
      "storageVendor": "...",
      "storageModel": "...",
      "storageSerialNumber": "..."
    },
    :
    :
  ]
}

```

レスポンスボディに含むデータフォーマットを次の表に示します。

表 4-11 HTM - Agent for RAID のインスタンスの一覧取得時のデータフォーマット

オブジェクト名	個数	説明
Collection	1	AgentForRAID オブジェクトを子に持つ Collection オブジェクト
AgentForRAID	0..n	AgentForRAID オブジェクト AgentForRAID オブジェクトのスキーマの詳細について、「表 4-12」に示します。

表 4-12 AgentForRAID オブジェクトのスキーマ

変数名	型	説明
instanceName	string	HTM - Agents のインスタンス名。
hostName	string	HTM - Agents が稼働しているホストの監視ホスト名。
ipAddr	string	HTM - Agents が稼働しているホストの IP アドレス。

変数名	型	説明
dataModelVersion	decimal	HTM - Agents のデータモデルバージョン。
lastUpdateTime ^{※1}	string	HTM - Agents の情報の更新処理が成功した時刻。 ISO8601 の拡張形式 (YYYY-MM-DDThh:mm:ssZ) で表します。hh は 24 時間表記です。また、z は UTC 時刻であることを表します。
storageVendor ^{※1}	string	ストレージシステム ^{※2} のベンダ名。
storageModel ^{※1}	string	ストレージシステム ^{※2} のモデル名。
storageSerialNumber ^{※1}	string	ストレージシステム ^{※2} の製造番号。

注※1

次の場合は null となります。

- 監視対象ストレージシステムが存在しない場合
- インスタンスの設定に不正がある場合
- PD レコードまたは PD_PDX レコードが収集されていない状態でエージェント情報をリフレッシュした場合
- 監視対象ストレージシステムが VSP Gx00 モデル、VSP Fx00 モデル、VSP E990、HUS VM、VSP 5000 シリーズ、VSP G1000、G1500、VSP F1500、Virtual Storage Platform シリーズのときに、パフォーマンスデータの収集にコマンドデバイスを使用していない場合

注※2

監視対象ストレージシステムには仮想ストレージマシンは含まれません。

HTM - Agent for RAID のインスタンス一覧取得で発生するエラー時のレスポンスについて次の表に示します。

表 4-13 HTM - Agent for RAID のインスタンス一覧取得時のエラーレスポンス

エラー発生条件	HTTP レスポンス	
	ステータスコード	Body に含まれるエラー ID
内部エラーが発生しました。	500	KATR00101-E

実行例 (HTM - Agent for RAID のインスタンス一覧の取得)

curl コマンドを使用したリクエストメッセージ:

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

リクエストに対する実行結果

リクエストヘッダー:

```
GET /TuningManager/v1/objects/AgentForRAID HTTP/1.1
Authorization: Basic c3lzZGVtOmlhbmFnZXI=
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: */*
```

レスポンスヘッダー:

```
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
```

レスポンスボディ:

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

4.6 パフォーマンスデータの取得

HTM - Agents が収集した監視対象のパフォーマンスデータを取得する方法を示します。なお、パフォーマンスデータは、CSV 形式で提供します。

HTTP リクエスト構文 (URI)

GET <scheme>://<host>:<port>/<application>/<version>/objects/<object-type>?<query-string>

補足

- objects ドメインの下の"<object-type>"は"<Agent 種別_レコード名>"の形式になっています。例えば"<object-type>"が"RAID_PD_RGC"の場合、HTM - Agent for RAID の PD_RGC レコードの情報を取得できます。
- <query-string> (クエリ文字列) には、<key>=<value>の形式で、キーとその値を入力します。<key>=<value>の区切りには&を使います。クエリ文字列の指定例を「クエリ文字列」に示します。また、クエリ文字列に指定可能なキーと値の型を「表 4-14 パフォーマンスデータ取得時に指定可能なクエリ文字列のキーと値」に示します。なお、クエリ文字列は Percent-Encoding が必要です。

クエリ文字列

クエリ文字の指定例：

```
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
```

表 4-14 パフォーマンスデータ取得時に指定可能なクエリ文字列のキーと値

キー	値		指定要否
	値域	概要	
hostName	ホスト名	HTM - Agents のホスト名	必須
agentInstanceName	HTM - Agents のインスタンス名	HTM - Agents のインスタンスを一意に識別できる値	必須
startTime ^{※1}	次の規則で指定します。(ISO8601 の拡張表記) 例：startTime=2013-11-20T19:54Z Value = <Year>-<Month>-<Day>T<Hour>:<Minute>Z ^{※2} それぞれの指定できる値について、説明します。 ・ Year : 0000~9999 ・ Month : 01~12 ・ Day : 01~31 ・ Hour : 00-23 ・ Minute : 00-59 なお、存在しない日付を指定した場合は、エラーとなります。	取得するパフォーマンスデータの開始時刻 (UTC)	任意
endTime ^{※1}	startTime と同じ規則です。 例：endTime=2013-11-21T19:54Z	取得するパフォーマンスデータの終了時刻 (UTC)	任意
fields	次の規則で指定します。 Value = 1*VCHAR / Value "%1F"Value; ^{※2} なお、複数指定 ^{※3} する場合、%1F ^{※4} 区切りとします。	レコードのフィールド名 各レコードのフィールド名については、マニュアル「Hitachi Command Suite Tuning Manager - Agents」のレコードについての説明を参照のこと。	任意
<レコードの各フィールド名>	次の規則で指定します。 Value= *VCHAR ; ^{※2} ^{※5} なお、値を複数指定 ^{※3} する場合、%1F ^{※4} 区切りとします。	キーにはフィールド名、値にはキーに指定したフィールドに対する値	任意
granularity ^{※6} , ^{※7}	指定できる値を次に示します。 ・ 未集約データを取得する場合：raw ・ 時間単位の要約データを取得する場合：hour ・ 日単位の要約データを取得する場合：day ・ 週単位の要約データを取得する場合：week ^{※8} ・ 月単位の要約データを取得する場合：month ^{※8}	取得する性能情報の時間粒度	任意

キー	値		指定要否
	値域	概要	
	<ul style="list-style-type: none"> 年単位の要約データを取得する場合： year^{※8} 指定しない場合は、raw を指定したと見なします。 PD レコードタイプのレコードの場合、要約データは存在しないため、raw を指定するか、または値を指定しないでください。 なお、大文字小文字は区別されません。		
accessMode ^{※6, ※7}	指定できる値を次に示します。 <ul style="list-style-type: none"> Timeline 方式でパフォーマンスデータを取得する場合：t Snapshot 方式でパフォーマンスデータを取得する場合：s 指定しない場合は、t を指定したと見なします。 なお、大文字小文字は区別されません。 補足 t を指定した場合でも、Timeline 方式でパフォーマンスデータを取得する条件を満たさないときには、Snapshot 方式でパフォーマンスデータを取得します。 ^{※9}	クエリを実行する際の取得方法 Hybrid Store で運用しているエージェントについて、Tuning Manager API でデータを取得する場合の取得方式には、次の2つの取得方式があります。 データを取得する際に指定する取得対象リソース、および取得対象期間に応じて、適切な取得方法を accessMode に指定することで、データの取得性能が向上します。 Timeline 方式 監視対象のリソースの中から特定のリソースに限定して、ある一定期間のパフォーマンスデータを取得したい場合に指定してください。特定のリソースに着目して、グラフを描画したい場合などに有効な取得方式です。 Snapshot 方式 監視対象のリソースのパフォーマンスデータを一括取得したい場合に指定してください。取得したデータをすべてデータウェアハウスに一括格納する場合などに有効な取得方式です。	任意

注※1

startTime と endTime を指定する場合、次の点に注意してください。

- startTime および endTime は、両方指定するか、または両方指定しないかのどちらかにしてください。一方だけを指定することはできません。
 なお、startTime および endTime の両方とも指定がない場合は、最新日時のパフォーマンスデータ（1 スナップショット分）だけが返されます。
- startTime は endTime 以下の値にしてください。

注※2

ABNF 表記です。

注※3

複数指定可能なキーについては、値を%1Fで区切るほかに、同じキー名を指定することで複数指定でき、意味も同じになります。また、1リクエスト内にこれらの表記を混在した指定もできます。

複数指定の例を次に示します。

- 値を%1Fで区切る場合：
fields=WRITE_IO_COUNT%1FWRITE_XFER_RATE
LDEV_NUMBER=00:00:01%1F00:00:02
- 同じキー名を指定する場合：
fields=WRITE_IO_COUNT&fields=WRITE_XFER_RATE
LDEV_NUMBER=00:00:01&LDEV_NUMBER=00:00:02

注※4

%1Fは、ASCIIコードのUnit Separator（印字不可能文字）です。

注※5

小数点を含む型（doubleおよびfloat）を選択条件の値として指定する場合、WebサーバやWebクライアントのロケールに関係なく、小数点にはピリオド（"."）を使って指定してください。小数点にコンマ（",”）を使って指定するとエラーになります。

注※6

Hybrid Store を使用して運用しているエージェントのパフォーマンスデータを取得している場合にだけ指定できます。

注※7

パフォーマンスデータの保持期間が過ぎてから削除されるタイミングが、Timeline方式のデータかSnapshot方式のデータかで異なるため、granularityおよびaccessModeの指定によって取得できるデータが異なることがあります。

注※8

監視対象ストレージシステムがVSP Gx00モデル、VSP Fx00モデル、VSP E990、HUS VM、VSP 5000シリーズ、VSP G1000、G1500、VSP F1500、およびVirtual Storage Platformシリーズの場合で、TCP/IP接続を使用して収集できるパフォーマンスデータを取得するときは、CSVデータのヘッダーに付与されるメタ情報だけが出力されます。

注※9

次のパフォーマンスデータについてはTimeline方式のデータが存在しないため、"t"を指定してもSnapshot方式でパフォーマンスデータを取得します。

- リストアまたはコンバートによって復元した期間のパフォーマンスデータ（復元した後に通常稼働してから収集した期間のパフォーマンスデータを同じ条件で参照するときと比較して、応答時間が遅くなります）
- TCP/IP接続を使用して収集できるパフォーマンスデータ（VSP Gx00モデル、VSP Fx00モデル、VSP E990、HUS VM、VSP 5000シリーズ、VSP G1000、G1500、VSP F1500、およびVirtual Storage Platformシリーズの場合）

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

レスポンスボディの構造を次に示します。

```

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 ;

```

レスポンスボディには CSV 形式のパフォーマンスデータが入ります。レスポンスボディの変数について次の表に示します。

表 4-15 レスポンスボディの変数

変数名	詳細
<SelectedFieldName_m>	指定フィールドの名称。※1
<FieldDelimiter>	フィールド区切り。 RFC4180 に従いコンマを指定します。(0x09)
<RecordDelimiter>	データ行の区切り。 データの末尾には、データの最後であることを示すために、もう 1 つ RecordDelimiter が入ります。 RFC4180 に従い CRLF (0x0D 0x0A) を使用します。
<DataType_m>	指定フィールドのデータ型。※1
<SelectedFieldValue_lm>	取得情報選択機能で選択した指定フィールドの値。※1, ※2, ※3, ※4

注※1

m は選択したフィールド番号です。

注※2

l は選択結果の行番号です。

注※3

小数点を含む型 (double および float) の値を取得した場合は、Web サーバや Web クライアントのロケールに関係なく、小数点はピリオド (".") になります。

注※4

Hybrid Store で運用している場合、データモデルがバージョンアップして既存のレコードに新しいフィールドが追加されたとき、追加されたフィールドには空文字 (長さが 0 バイトの文字列) が出力されます。

パフォーマンスデータ取得で発生するエラー時のレスポンスについて次の表に示します。

表 4-16 パフォーマンスデータ取得時のエラーレスポンス

エラー発生条件	HTTP レスポンス			
	ステータスコード	ヘッダー		Body に含まれるエラー ID
		Name	Value	
指定したホストの Tuning Manager Agent REST API コンポーネントにアクセスできましたが、指定したインスタンスの Agent 種別の情報が取得できません。	404	追加されるヘッダーはありません。		KATR12054-E
指定したインスタンスを Tuning Manager Server が認識していません。	404			KATR12005-E
指定したインスタンスがエージェントホスト上で削除されています。	404			KATR11010-E

エラー発生条件	HTTP レスポンス			Body に含まれるエラー ID
	ステータスコード	ヘッダー		
		Name	Value	
指定したエージェントインスタンスが稼働するホストに対するリクエストが集中しています。	503	Retry-After	60	KATR12001-W
存在しない、またはサポートしていないレコード種別を指定しました。	404	追加されるヘッダーはありません。		KATR11012-E
リクエスト不正です。	400			次のどれかです。 <ul style="list-style-type: none"> • KATR11008-E • KATR11014-E • KATR11016-E • KATR11018-E • KATR11020-E • KATR12006-E • KATR12009-E
HTM - Agents で Tuning Manager API を使うための設定が有効になっていません。	500			KATR12004-E
HTM - Agents のバージョンが、実行している API バージョンに対応していません。	500			KATR12003-E
Tuning Manager Server と HTM - Agents 間の通信エラー（接続拒否）が発生しました。	500			KATR12007-E
Tuning Manager Server と HTM - Agents 間の通信エラー（接続不可）が発生しました。	500			KATR12008-E
HTM - Agents から、403 以外のステータスコード、かつ text/csv または application/json 以外のメディアタイプが返されました。	500			KATR12048-E
内部矛盾（Tuning Manager Server）が発生しました。	500			次のどれかです。 <ul style="list-style-type: none"> • KATR00101-E • KATR00103-E • KATR12002-E
内部矛盾（HTM - Agents）が発生しました。	500			KATR11022-E
Agent インスタンスの初期化処理中です。	503			KATR11024-E
レスポンス返却中の内部エラーが発生しました。	レスポンスの返却を中止します。			

実行例

- 実行例 1（HTM - Agent for RAID の PD レコードの情報を取得する場合）
- 実行例 2（HTM - Agent for RAID の PL_LDS レコードの情報を取得する場合）
- 実行例 3（HTM - Agent for RAID の PD_PTC レコードの情報取得）
- 実行例 4（HTM - Agent for RAID の PL_PRCS レコードの情報を startTime および endTime を指定して取得する場合）

実行例 1 (HTM - Agent for RAID の PD レコードの情報を取得する場合)

curl コマンドを使用したリクエストメッセージ:

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

リクエストに対する実行結果

リクエストヘッダー:

```
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: */*
```

レスポンスヘッダー:

```
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
```

レスポンスボディ:

```
DKC_SERIAL_NUMBER,VENDOR_ID,DKC_NAME,CACHE_MEMORY_CAPACITY,CACHE_MEMORY_I
NSTALLED_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,stin
g(8),time_t
"10051","HITACHI","VSP G1000",469504,0,"",2014-06-20 13:00:07,3600,"PD",
2014-06-20 13:00:07
```

実行例 2 (HTM - Agent for RAID の PI_LDS レコードの情報を取得する場合)

curl コマンドを使用したリクエストメッセージ:

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

リクエストに対する実行結果

リクエストヘッダー:

```
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: */*
```

レスポンスヘッダー:

```
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
```


X-Pad: avoid browser bug

レスポンスボディ :

```
PORT_NUMBER, PORT_NAME, CHA_NAME, PORT_WWN, PORT_TYPE, PORT_ROLE, PORT_SPEED, SI
PR_NUMBER, COLLECTION_TIME_INTERVAL, INPUT_RECORD_TYPE, RECORD_TIME
string(8), string(64), string(16), string(32), string(8), string(16), string(8)
, string(8), time_t, ulong, string(8), time_t
"0", "CL1-A", "CHA-1EU", "50060E8007274300", "FIBRE", "Target", "Auto", "",
2014-06-20 12:00:07, 3600, "PTC", 2014-06-20 12:00:07
"1", "CL1-B", "CHA-1EU", "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
```

実行例 4 (HTM - Agent for RAID の PI_PRCS レコードの情報を startTime および endTime を指定して取得する場合)

curl コマンドを使用したリクエストメッセージ :

```
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
```

リクエストに対する実行結果

リクエストヘッダー :

```
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: */*
```

レスポンスヘッダー :

```
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
```

レスポンスボディ :

```
PROCESSOR_ID, ADAPTOR_ID, CONTROLLER, PROCESSOR_TYPE, PROCESSOR_BUSY_RATE, MAX
_PROCESSOR_BUSY_RATE, MAX_BUFFER_LENGTH, BUFFER_IO_COUNT, MAX_BUFFER_IO_COUN
T, BUFFER_IO_RATE, MAX_BUFFER_IO_RATE, INTERVAL, INPUT_RECORD_TYPE, RECORD_TIM
E
string(16), string(16), string(8), string(8), float, float, float, float, float, f
loat, 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
```

その他の Tuning Manager API

この章では、アラート機能やパフォーマンスデータの取得に関連しない API について、説明します。

なお、実行例には見やすくするために改行を入れています。

- [5.1 Tuning Manager API のリソース一覧](#)
- [5.2 アラート定義情報の設定および運用のアクション一覧取得](#)
- [5.3 HTM - Agents 管理のアクション一覧の取得 \(アラート機能の場合\)](#)
- [5.4 HTM - Agents 管理のアクション一覧の取得 \(パフォーマンスデータの取得の場合\)](#)
- [5.5 API の詳細バージョンの情報取得 \(アラート機能の場合\)](#)
- [5.6 API の詳細バージョンの情報取得 \(パフォーマンスデータの取得の場合\)](#)
- [5.7 メタ情報取得 \(パフォーマンスデータの取得の場合\)](#)

5.1 Tuning Manager API のリソース一覧

Tuning Manager API のリソース一覧を次の表に示します。なお、リソースは URI で表記していません。

表 5-1 Tuning Manager API のリソース一覧

API の用途	リクエスト		参照先
	HTTP method	URI	
アラート定義情報の設定および運用のアクション一覧取得	GET	/<application>/v1/services/alert-def-manager/actions	5.2
HTM - Agents 管理のアクション一覧取得 (アラート機能の場合)	GET	/<application>/v1/services/agent-manager/actions	5.3
HTM - Agents 管理のアクション一覧の取得 (パフォーマンスデータの取得の場合)	GET	/<application>/v1/services/AgentManager/actions	5.4
API の詳細バージョンの情報取得 (アラート機能の場合)	GET	/<application>/v1/configuration/version	5.5
API の詳細バージョンの情報取得 (パフォーマンスデータの取得の場合)	GET	/<application>/v1/configuration/Version	5.6
Collection オブジェクトのメタ情報の取得	GET	/<application>/v1/domain-types/Collection	5.7
Error オブジェクトのメタ情報の取得	GET	/<application>/v1/domain-types/Error	5.7
Version オブジェクトのメタ情報の取得	GET	/<application>/v1/domain-types/Version	5.7
Action オブジェクトのメタ情報の取得	GET	/<application>/v1/domain-types/Action	5.7
Job オブジェクトのメタ情報の取得	GET	/<application>/v1/domain-types/Job	5.7
Agents オブジェクトのメタ情報の取得	GET	/<application>/v1/domain-types/Agents	5.7
AgentForRAID オブジェクトのメタ情報の取得	GET	/<application>/v1/domain-types/AgentForRAID	5.7

5.2 アラート定義情報の設定および運用のアクション一覧取得

アラート定義情報の設定および運用のアクションの取得方法について示します。

HTTP リクエスト構文 (URI)

GET <scheme>://<host>:<port>/<application>/<version>/services/alert-def-manager/actions

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

レスポンスボディの構造を次に示します。

```

{
  "data": [
    {
      "name": "<アクション名>",
      "url": "<アクションを実行する URI>",
      "method": "<アクション実行時の HTTP メソッド>",
      "type": "<アクション実行時のリクエストの Content-Type ヘッダに指定するメディアタイプ>"
    }
  ]
}

```

実行例（アラート定義情報の設定および運用のアクション一覧の取得）

curl コマンドを使用したリクエストメッセージ：

```

curl -v -H "Content-Type: application/json" -u system:manager -X GET
http://server_1:22015/DeviceManagerWebService/analytics/v1/services/alert-def-manager/actions

```

リクエストに対する実行結果／

リクエストヘッダー：

```

GET /DeviceManagerWebService/analytics/v1/services/alert-def-manager/actions HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22015
Accept: */*
Content-Type: application/json

```

レスポンスヘッダー：

```

HTTP/1.1 200 OK
Date: Tue, 20 Oct 2015 07:34:13 GMT
Server: Cosminexus HTTP Server
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json; charset=UTF-8

```

レスポンスボディ：

```

{
  "data": [
    {
      "name": "sync",
      "url": "http://server_1:22015/DeviceManagerWebService/analytics/v1/services/alert-def-manager/actions/sync/invoke",
      "method": "PUT",
      "type": "application/json"
    }
  ]
}

```

5.3 HTM - Agents 管理のアクション一覧の取得（アラート機能の場合）

HTM - Agents 管理が管理するアラート機能のアクション一覧の取得方法について示します。

HTTP リクエスト構文 (URI)

GET <scheme>://<host>:<port>/<application>/<version>/services/agent-manager/actions

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

レスポンスボディの構造を次に示します。

```
{
  "data": [
    {
      "name": "<アクション名>",
      "url": "<アクションを実行する URI>",
      "method": "<アクション実行時の HTTP メソッド>",
      "type": "<アクション実行時のリクエストの Content-Type ヘッダに指定するメディアタイプ>"
    }
  ]
}
```

実行例 (HTM - Agents 管理のアラート機能のアクション一覧の取得)

curl コマンドを使用したリクエストメッセージ:

```
curl -v -H "Content-Type: application/json" -u system:manager -X GET
http://server_1:22015/DeviceManagerWebService/analytics/v1/services/agent-manager/actions
```

リクエストに対する実行結果

リクエストヘッダー:

```
GET /DeviceManagerWebService/analytics/v1/services/agent-manager/actions
HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22015
Accept: */*
Content-Type: application/json
```

レスポンスヘッダー:

```
HTTP/1.1 200 OK
Date: Tue, 20 Oct 2015 07:34:14 GMT
Server: Cosminexus HTTP Server
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json;charset=UTF-8
```

レスポンスボディ:

```
{
  "data": [
    {
      "name": "refresh",
      "url": "http://server_1:22015/DeviceManagerWebService/analytics/v1/services/agent-manager/actions/refresh/invoke",
      "method": "PUT",
      "type": "application/json"
    }
  ],
}
```

```

    {
      "name": "list",
      "url": "http://server_1:22015/DeviceManagerWebService/
analytics/v1/services/agent-manager/actions/list/invoke",
      "method": "PUT",
      "type": "application/json"
    }
  ]
}

```

5.4 HTM - Agents 管理のアクション一覧の取得（パフォーマンスデータの取得の場合）

HTM - Agents 管理のアクション一覧の取得方法について示します。

この API は、HTM - Agents 管理が提供するすべてのアクションについて、parameters フィールドが省略された Action オブジェクトを Collection オブジェクトに格納して、API クライアントに返します。この Collection オブジェクト（アクション一覧）はどのようなアクションが存在するかを知るために使用できます。

HTTP リクエスト構文 (URI)

GET <scheme>://<host>:<port>/<application>/<version>/services/AgentManager/actions

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

レスポンスボディの構造を次に示します。

```

{
  "items": [{"name": "<アクション名>",
    "url": "<アクションを実行する URI>",
    "method": "<アクション実行時の HTTP メソッド>",
    "type": "<アクション実行時のリクエストの Content-Type ヘッダに指定するメディアタイプ>"
  }]
}

```

レスポンスボディに含むデータフォーマットを次の表に示します。

表 5-2 HTM - Agents 管理のアクション一覧取得のデータフォーマット

オブジェクト名	個数	説明
Collection	1	Action オブジェクトを子に持つ Collection オブジェクト。
- Action	0...n	Action オブジェクト。 ただし、parameters 変数はありません。

実行例 (HTM - Agents 管理のアクション一覧の取得)

curl コマンドを使用したリクエストメッセージ:

```

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

```

リクエストに対する実行結果 /

リクエストヘッダー :

```
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
```

レスポンスヘッダー :

```
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
```

レスポンスボディ :

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

5.5 API の詳細バージョンの情報取得（アラート機能の場合）

API の詳細バージョンの情報取得の性質を次の表に示します。

HTTP リクエスト構文 (URI)

```
GET <scheme>://<host>:<port>/<application>/<version>/configuration/
version
```

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

レスポンスボディの構造を次に示します。

```
{
  "productName": "<製品名>"
  "apiVersion": "<API バージョン>",
}
```

表 5-3 Version オブジェクトのスキーマ（アラート機能の場合）

変数名	型	説明
productName	string	Tuning Manager server の製品名。 つまり, "Hitachi Tuning Manager"になります。
apiVersion	string	API のバージョン。 「<メジャーバージョン>.<マイナーバージョン>.<リビジョン>」の形式で表します。

実行例 (API の詳細バージョンの情報取得)

curl コマンドを使用したリクエストメッセージ:

```
curl -v -H "Content-Type: application/json" -u system:manager -X GET
http://server_1:22015/DeviceManagerWebService/analytics/v1/configuration/
version
```

リクエストに対する実行結果:

リクエストヘッダー:

```
GET /DeviceManagerWebService/analytics/v1/configuration/version HTTP/1.1
Authorization: Basic c3lzdGVtOmlhbmFnZXI=
User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/
3.13.1.0 zlib/1.2.3 libidn/1.18 libssh2/1.2.2
Host: server_1:22015
Accept: */*
Content-Type: application/json
```

レスポンスヘッダー:

```
HTTP/1.1 200 OK
Date: Tue, 20 Oct 2015 07:34:13 GMT
Server: Cosminexus HTTP Server
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Content-Type: application/json; charset=UTF-8
```

レスポンスボディ:

```
{
  "productName": "Hitachi Tuning Manager",
  "apiVersion": "1.0.0",
}
```

5.6 API の詳細バージョンの情報取得 (パフォーマンスデータの取得の場合)

API の詳細バージョンの情報取得の性質を次の表に示します。

HTTP リクエスト構文 (URI)

```
GET <scheme>://<host>:<port>/<application>/<version>/configuration/
Version
```

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

レスポンスボディの構造を次に示します。

```
{
  "productName": "<製品名>",
  "productVersion": "<プロダクトバージョン>",
  "apiVersion": "<API バージョン>",
  "description": "<API バージョンの説明>"
}
```

表 5-4 Version オブジェクトのスキーマ（パフォーマンスデータの取得の場合）

変数名	型	説明
productName	string	Tuning Manager server の製品名。 つまり, "Hitachi Tuning Manager"になります。
productVersion	string	Tuning Manager server のプロダクトバージョン。 つまり, Tuning Manager server のプロダクトバージョンが v8.0.0 の場合, "8.0.0" になります。
apiVersion	string	API のバージョン。 「<メジャーバージョン>.<マイナーバージョン>」の形式で表します。
description	string	API のバージョンについての説明。

実行例（APIの詳細バージョンの情報取得）

curl コマンドを使用したリクエストメッセージ:

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

リクエストに対する実行結果

リクエストヘッダー:

```
GET /TuningManager/v1/configuration/Version 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
```

レスポンスヘッダー:

```
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
```

レスポンスボディ:

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

5.7 メタ情報取得（パフォーマンスデータの取得の場合）

パフォーマンスデータリソースを除く、情報取得 API で使用する JSON Object に関するメタ情報を取得する際に使用します。メタ情報は JSON Schema に従って表現します。

なお、パフォーマンスデータリソースのメタ情報は CSV データのヘッダーに付与して提供し、メタ情報だけでは取得できません。

HTTP リクエスト構文 (URI)

GET <scheme>://<host>:<port>/<application>/<version>/domain-types/
<object-type>

補足

domain-types ドメインの下の<object-type>は、次の形式を指定してください。

- Collection オブジェクトのメタ情報を取得する場合：Collection
- Error オブジェクトのメタ情報を取得する場合：Error
- Version オブジェクトのメタ情報を取得する場合：Version
- Action オブジェクトのメタ情報を取得する場合：Action
- Job オブジェクトのメタ情報を取得する場合：Job
- Agents オブジェクトのメタ情報を取得する場合：Agents
- AgentForRAID オブジェクトのメタ情報を取得する場合：AgentForRAID

リクエストの構造

リクエストボディは、空でなければいけません。

レスポンスの構造

スキーマは各オブジェクトを表す JSON Schema に従います。

AgentForRAID オブジェクトのメタ情報取得時のレスポンスボディの構造を次に示します。

```
{  "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" ] }
```

メタ情報取得で発生するエラー時のレスポンスについて次の表に示します。

表 5-5 メタ情報取得時のエラーレスポンス

エラー発生条件	HTTP レスポンス		
	ステータス コード	ヘッダー	Body に含まれるエラー ID
		Name	Value
存在しない、またはサポートしていないメタ情報の種別を指定しました。	404	追加されるヘッダーはありません。	
内部矛盾が発生しました。	500	KATR00101-E	
レスポンス返却中の内部エラーが発生しました。	レスポンスの返却を中止します。		

実行例 (AgentForRAID オブジェクトのメタ情報取得)

curl コマンドを使用したリクエストメッセージ：

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

リクエストに対する実行結果

リクエストヘッダー :

```
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: */*
```

レスポンスヘッダー :

```
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
```

レスポンスボディ :

```
{  "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" ]}
```

Tuning Manager API を実行するコマンド

この章では、Tuning Manager API を実行するコマンドとその文法について説明します。

Tuning Manager API を実行するコマンドの記述形式および文法規則は、Windows および Linux で共通です。Linux の場合、制御端末からコマンドを実行します。

- [6.1 コマンドの使用法](#)
- [6.2 コマンド実行に必要な設定](#)
- [6.3 コマンド実行の前提条件](#)
- [6.4 htmrest](#)

6.1 コマンドの使用法

CLI の一般的な使用法を次に示します。

- OS のコマンドラインで CLI コマンドを手動で実行する。
- スクリプト、マクロおよび開発製品内で CLI コマンドを呼び出す。

Perl, Microsoft Visual Basic, Tcl は、この目的に使用される最も一般的なツールです。スクリプトなどを使用すると、データを自動的に報告し、集積できます（例：スケジュール設定されたスクリプトの実行、結果の解析／操作によってカスタマイズされた報告）。

注意

Windows Server 2008 または Windows Server 2012 で UAC (User Account Control) 機能が有効に設定されている場合、Tuning Manager シリーズでは、管理者として実行しているコマンドプロンプトでコマンドを実行することを推奨します。詳細については、マニュアル「Hitachi Command Suite Tuning Manager インストールガイド」を参照してください。

コマンドの指定形式を次に示します。

図 6-1 コマンドの指定形式

```
xxxxxx [-option-A [value-a [, value-b [, value-c...]]] ... (1)
        [-option-B [value-a [, value-b [, value-c...]]] ... (1) } ... (2)
        [arbitrary-name-X[arbitrary-name-Y[arbitrary-name-Z...]]]
```

(1) を「オプション」と呼びます。(2) を「引数」と呼びます。

コマンドの文法の説明に使用する記号

コマンドの文法の説明に使用する記号を次の表に示します。

表 6-1 コマンドの文法の説明に使用する記号

記号	意味と例
 (ストローク)	複数の項目に対して項目間の区切りを示し、「または」の意味を示します。 (例) 「A B C」は、「A, B, または C」を示します。
{ } (波括弧)	この記号で囲まれている複数の項目の中から、必ず一組の項目を選択します。項目と項目の区切りは「 」で示します。 (例) 「{A B C}」は、「A, B, または C のどれかを必ず指定する」ことを示します。
[] (角括弧)	この記号で囲まれている項目は、任意に指定できます (省略できます)。 (例) 「[A]」は、「必要に応じて A を指定する」ことを示します (必要でない場合は、A を省略できます)。 「[B C]」は、「必要に応じて B, または C を指定する」ことを示します (必要でない場合は、B および C を省略できます)。
... (点線)	この記号の直前に示された項目を繰り返して複数個、指定できます。なお、項目を複数個指定する場合は、項目の区切りに 1 バイトの空白文字 (半角スペース) を使用します。 (例) 「A B...」は、「A の後ろに、B を複数個指定できる」ことを示します。

6.2 コマンド実行に必要な設定

6.2.1 Java 環境の設定

Tuning Manager API を実行するコマンドを使用するには、Java 環境を設定する必要があります。Tuning Manager API を実行するコマンドの Java 環境を設定するには、Device Manager がサポートしているバージョンの Java が実行マシンにインストールされていることが前提です。その上で、Tuning Manager API を実行するコマンドの実行環境に応じて適切な JRE を指定します。

JRE は、次のどちらかの方法で指定します。

- 環境変数 `HTMREST_CLI_JRE_PATH` による指定（推奨）

Hitachi Command Suite 製品同梱の JRE^{※1}、または Oracle 社の Web サイトからダウンロードした JRE を指定できます。

- 環境変数 `PATH` による指定^{※2}

Oracle 社の Web サイトからダウンロードした JRE を指定してください。

注※1

SSL で接続する場合、Hitachi Command Suite 製品同梱の JRE は指定しないでください。

Hitachi Command Suite 製品同梱の JRE を指定すると、次のようなメッセージが出力されます。メッセージが出力されたときは、Oracle 社の Web サイトからダウンロードした JRE に指定し直してください。

```
Connection error (detailed information: java.net.SocketException:
java.lang.ClassNotFoundException:
com.hitachi.truenorth.HiCommand.Base.Common.HBaseSSLSocketFactory).
```

注※2

JRE のパスの指定に環境変数 `PATH` を使用する場合、次の点に注意してください。

- Hitachi Command Suite 製品同梱の JRE は指定しないでください。
- Java を使用しているほかのアプリケーションの動作に影響を与えるおそれがあります。
- JRE が自動更新される環境の場合、JRE のバージョンが Device Manager がサポートしていないバージョンのものに意図しないで更新されるおそれがあります。

参考

Hitachi Command Suite 製品同梱の JRE は、次の場所に格納されています。

Windows の場合

```
<Hitachi Command Suite のインストール先フォルダ>%Base64%uCPSB\jdk\jre\bin
```

Linux の場合

```
<Hitachi Command Suite のインストール先ディレクトリ>/Base64/
uCPSB/jdk/jre/bin
```

6.2.2 SSL の設定

Tuning Manager API を実行するコマンドでは、Tuning Manager server と Device Manager を同じホストにインストールしている場合、HBase 64 Storage Mgmt Web Service に SSL を設定することで通信データを暗号化できます。

注意

Tuning Manager server と Device Manager を別々のホストにインストールしている場合、Tuning Manager API を実行するコマンドの通信データは暗号化できません。

1. Tuning Manager server と Device Manager をインストールしているホストの HBase 64 Storage Mgmt Web Service に SSL を設定します。

設定方法については、マニュアル「Hitachi Command Suite システム構成ガイド」の SSL サーバの構築（Hitachi Command Suite 共通コンポーネント）について説明している個所を参照してください。

2. Tuning Manager server と Device Manager をインストールしているホストの次のトラストストアに、サーバ証明書をインポートします。

Windows の場合：

Oracle 社の Web サイトからダウンロードした JRE が C:\Program Files\Java\jdk1.8.0 フォルダにインストールされているときの設定例を次に示します。

```
C:\Program Files\Java\jdk1.8.0\jre\bin\keytool.exe -import -alias <エイリアス名> -trustcacerts -file <証明書のファイル名> -keystore <トラストストアファイル名> -storepass <トラストストアへのアクセスパスワード>
```

Linux の場合：

Oracle 社の Web サイトからダウンロードした JRE が /usr/Java/jdk1.8.0 ディレクトリにインストールされているときの設定例を次に示します。

```
/usr/Java/jdk1.8.0/bin/keytool -import -alias <エイリアス名> -trustcacerts -file <証明書のファイル名> -keystore <トラストストアファイル名> -storepass <トラストストアへのアクセスパスワード>
```

- alias：トラストストア内で証明書を識別するための名称を指定します。サーバ証明書が複数ある場合は、トラストストア内で使用されていない任意のエイリアス名を指定してください。
 - file：証明書を指定します。
 - keystore：インポート先のトラストストアファイルのパスを指定します。
3. 手順 2 で keystore オプションに指定したトラストストアファイルのパスを環境変数 HTMREST_CLI_CERTS_PATH に設定します。

なお、環境変数 HTMREST_CLI_CERTS_PATH を設定しなかった場合、次のパスがデフォルトで設定されます。

Windows の場合：

```
<Hitachi Command Suite のインストール先フォルダ>\DeviceManager\HiCommandServer\tools\htmrest\HtmRestCerts
```

Linux の場合：

```
<Hitachi Command Suite のインストール先ディレクトリ>/HiCommandServer/tools/htmrest/HtmRestCerts
```

JRE の環境変数については、「6.2.1 Java 環境の設定」を参照してください。

6.3 コマンド実行の前提条件

コマンド実行の前提条件を次に示します。

- Device Manager をインストールしているホストで実行する
- 実行する Tuning Manager API の実行条件を満たしている

Tuning Manager API の実行条件については、「1.4 API で監視対象のしきい値情報にアクセスできる条件」を参照してください。

- Device Manager をインストールしているホストで Java 環境が設定されている

Java 環境の設定については、「6.2.1 Java 環境の設定」を参照してください。

6.4 htmrest

形式

```
htmrest <HTTP メソッド> <実行する Tuning Manager API の URI>
-u | --user <ユーザー ID>
-p | --password <パスワード>
[ -b | --body <リクエストファイルのパス> ]
[ --connect-timeout <Tuning Manager API へ接続する処理のタイ
ムアウト値> ]
[ --read-timeout <Tuning Manager API から取得したデータの読み
取り処理のタイムアウト値> ]
```

機能

Device Manager をインストールしているホストから、Tuning Manager API を使用してアラート機能の運用やパフォーマンスデータの取得などを実行します。実行結果は、標準出力に出力します。

実行権限

権限は必要ありません。

格納先ディレクトリ

Windows の場合

```
<Hitachi Command Suite のインストール先フォルダ>%DeviceManager
%HiCommandServer%tools%htmrest%
```

Linux の場合

```
<Hitachi Command Suite のインストール先ディレクトリ>/HiCommandServer/tools/
htmrest/
```

オプション

htmrest コマンド引数のオプションを次の表に示します。

表 6-2 htmrest コマンド引数のオプション

オプション	必須 任意	説明
<HTTP メソッド>	必須	HTTP メソッド (GET, POST, PUT または DELETE) をどれか 1 つ指定します。
<実行する Tuning Manager API の URI>	必須	実行する API の URL を指定します。
-u --user <ユーザー ID>	必須	Hitachi Command Suite にログインするためのユーザー ID を指定します。
-p --password <パスワード>	必須	Hitachi Command Suite にログインするためのパスワードを指定します。
-b --body <リクエストファイルのパス>	POST および PUT メソッドの場合、必須	Tuning Manager API のリクエストボディに指定したい内容を記載したファイルのパスを指定します。 注意 <ul style="list-style-type: none">ファイルの記載内容は JSON 形式である必要があります。ファイルで使用できる文字コードは UTF-8 です。ファイルで使用できる改行コードは次のとおりです。

オプション	必須 任意	説明
		Windows の場合：CR+LF Linux の場合：LF 。 絶対パスまたは、コマンドを実行するディレクトリからの相対パスで指定します。
--connect-timeout <Tuning Manager API へ接続する処理のタイム アウト値>	任意	Tuning Manager API へ接続する処理のタイムアウト値を秒単位で指定します。指定できる範囲は、0～1000000 です。 ・ 省略した場合：300 秒 ・ 0 を入力した場合：タイムアウトされない
--read-timeout <Tuning Manager API から取得したデータの読 み取り処理のタイムアウ ト値>	任意	Tuning Manager API から取得したデータの読み取り処理のタイムアウト値を秒単位で指定します。指定できる範囲は、0～1000000 です。 ・ 省略した場合：3,600 秒 ・ 0 を入力した場合：タイムアウトされない

注意事項

- ・ このコマンドは、絶対パスでは実行されません。コマンドの格納先まで移動したあと、相対パスで実行してください。
- ・ このコマンドは多重実行できます。
- ・ コマンドから実行した Tuning Manager API のエラーについては、API のレスポンスで確認してください。

戻り値

表 6-3 htmrest コマンドの戻り値

戻り値	意味
0	Tuning Manager API 実行時のステータスコードが 200～299 である
1	オプションで指定した値が不正である
2	オプションで指定したファイルにアクセスできない
3	オプションで指定したファイルが変換できない
4	Tuning Manager API にアクセスできない
5	メモリーが不足している
50	Tuning Manager API 実行時のステータスコードが 200～299 以外である
99	予期しないエラーが発生した

実行例（アラート定義情報を参照する場合）

```
htmrest.bat GET http://server_1:22015/DeviceManagerWebService/
analytics/v1/objects/alert-defs -u system -p manager
```

出力例（アラート定義情報を参照する場合）

```
200
{"data":[{"alertDefName":"AlertVSPG1000_01"},
{"alertDefName":"AlertVSPG1000_02"}, {"alertDefName":"AlertVSPG1000_03"},
{"alertDefName":"AlertUSPV_01"}]}
```

コマンドの出力結果の 1 行目には API のステータスコードが、2 行目には API のレスポンスボディが出力されます。

API のメッセージ

この章では、API 実行時のメッセージについて説明します。

- 7.1 メッセージの形式
- 7.2 メッセージの出力先一覧
- 7.3 メッセージ一覧

7.1 メッセージの形式

API 実行時に出力するメッセージの形式と、この章での記載形式を示します。

7.1.1 メッセージの出力形式

API 実行時に出力するメッセージの形式を説明します。メッセージは、メッセージ ID とそれに続くメッセージテキストで構成されます。

記載形式の例を次に示します。

KATR または KAICnnnnn-Y<メッセージテキスト>

メッセージ ID は、次の内容を示しています。

- KATR : Tuning Manager API の利用時のメッセージであることを示します。
- KAIC : アラート機能の利用時のメッセージであることを示します。
- nnnnn : メッセージの通し番号を示します。
- Y : メッセージの種類を示します。
 - E : エラー
処理は中断されます。
 - W : 警告
メッセージ出力後、処理は続けられます。
 - I : 情報
ユーザーに情報を知らせます。

7.1.2 メッセージの記載形式

この章でのメッセージの記載形式を示します。メッセージテキストで斜体になっている部分は、メッセージが表示される状況によって表示内容が変わることを示しています。また、メッセージをメッセージ ID 順に記載しています。記載形式の例を次に示します。

メッセージ ID *メッセージテキスト* *メッセージの説明文*

(S) システムの処置を示します。

(O) メッセージが表示されたときに、オペレーターがとる処置を示します。

7.2 メッセージの出力先一覧

ここでは、API 実行時に出力する各メッセージの出力先を一覧で示します。

表 7-1 API 実行時に出力するメッセージの出力先一覧 (Tuning Manager server ホスト)

出力先	メッセージ ID
• <i>メッセージログ</i>	KATR10001, KATR10002, KATR10005, KATR10007~KATR10010, KATR10012, KATR10015, KATR10017, KATR10018, KATR10020, KATR10041~KATR10044, KATR11003~KATR11007, KATR11028~KATR11031
• <i>Error オブジェクト</i> ※	KATR00102, KATR10006, KATR10011, KATR10013, KATR10016

出力先	メッセージ ID
<ul style="list-style-type: none"> Error オブジェクト※ メッセージログ 	KATR00101, KATR00103, KATR00105~KATR00113, KATR10019, KATR10040, KATR10045, KATR11002, KATR11032, KATR11037, KATR11038, KATR12001~KATR12009, KATR12048, KATR12054

注※

Error オブジェクトは API クライアントに返されます。

表 7-2 API 実行時に出力するメッセージの出力先一覧 (Agent ホスト)

出力先	メッセージ ID
<ul style="list-style-type: none"> メッセージログ 	KATR10046~KATR10051, KATR11026, KATR11027, KATR12010, KATR12014, KATR12025~KATR12030, KATR12040, KATR12041, KATR12044, KATR12046, KATR12047, KATR12049~KATR12053, KATR15002~KATR15022
<ul style="list-style-type: none"> Error オブジェクト※ メッセージログ 	KATR00105, KATR11008~KATR11025, KATR11033~KATR11034

注※

Error オブジェクトは API クライアントに返されます。

7.3 メッセージ一覧

「表 7-3 API が出力するメッセージ」では、API が出力するメッセージと対処方法について説明します。

表 7-3 API が出力するメッセージ

メッセージ ID	メッセージテキスト	メッセージの説明文
KATR00101-E	An unexpected error occurred. (<処理の名称 ({ライセンス認証処理 エージェント情報のリフレッシュ処理 エージェント情報の取得処理 パフォーマンスデータの取得処理 メタ情報取得処理}) >) 予期しないエラーが発生しました。 (<処理の名称 ({ライセンス認証処理 エージェント情報のリフレッシュ処理 エージェント情報の取得処理 パフォーマンスデータの取得処理 メタ情報取得処理}) >)	予期しないエラーが発生しました。 (S) 処理を中断し、エラーレスポンスを返します。 (O) システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。
KATR00102-E	An unexpected error occurred. (<処理の名称 ({ライセンス認証処理 ユーザー認証処理}) >) 予期しないエラーが発生しました。 (<処理の名称 ({ライセンス認証処理 ユーザー認証処理}) >)	予期しないエラーが発生しました。 (S) 処理を中断し、エラーレスポンスを返します。 (O) Tuning Manager のサーバ管理者に連絡してください。
KATR00103-E	A database access error occurred. (<処理の名称 ({エージェント情報のリフレッシュ処理 エージェント情報の取得処理 パフォーマンスデータの取得処理 ライセンス認証処理}) >)	データベースアクセス中にエラーが発生しました。 (S) 処理を中断し、エラーレスポンスを返します。 (O)

メッセージID	メッセージテキスト	メッセージの説明文
	データベースアクセスエラーが発生しました。(＜処理の名称(エージェント情報のリフレッシュ処理 / エージェント情報の取得処理 / パフォーマンスデータの取得処理 / ライセンス認証処理)＞)	Tuning Manager Server を再起動してください。再度エラーが発生する場合は、システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口へ連絡してください。
KATR00105-E	The specified HTTP method is not allowed. (allowed methods =<使用可能な HTTP メソッド>) 許可されていない HTTP メソッドが指定されています。許可されている HTTP メソッド:<使用可能な HTTP メソッド>	許可していない HTTP メソッドでアクセスしました。 (S) 処理を中断します。 (O) HTTP メソッドを確認してください。
KATR00106-E	An unexpected error occurred during initialization of Tuning Manager REST Application Server. Initialization failed. (maintenance information =<エラーの詳細情報>) Tuning Manager REST Application Server の初期化で予期せぬエラーが発生し、初期化に失敗しました。詳細情報=<エラーの詳細情報>	Tuning Manager REST Application Server の初期化に予期せぬエラーが発生しました。 (S) 初期化を中止します。サービスは停止しません。 (O) Tuning Manager Server を再起動してください。再度エラーが発生する場合は、システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口へ連絡してください。
KATR00107-E	The request message does not include all the required headers. (missing header =<不足している必須ヘッダ>) リクエストに必須のヘッダが含まれていませんでした。不足していたヘッダ=<不足している必須ヘッダ>	リクエストに必須ヘッダが含まれていません。 (S) 処理を中断し、エラーレスポンスを返します。 (O) ヘッダを確認してください。
KATR00108-E	No resource exists at the specified URL. (URL =<URL>) 指定された URL は存在しませんでした。URL=<URL>	存在しない URL が指定されています。 (S) 処理を中断し、エラーレスポンスを返します。 (O) URL を確認してください。
KATR00109-E	The media type specified in the Accept header is unsupported. Accept ヘッダで指定されたメディアタイプは受け付けられません。	Accept ヘッダで指定しているメディアタイプは、本 URL では返すことができないタイプです。 (S) 処理を中断し、エラーレスポンスを返します。 (O) Accept ヘッダを確認してください。
KATR00110-E	The encoding format specified in the Accept-Encoding header is unsupported. Accept-Encoding ヘッダで指定されたエンコーディング形式は受け付けられません。	Accept-Encoding ヘッダで指定しているエンコーディング形式は、Tuning Manager Server がサポートしていない形式です。 (S) 処理を中断し、エラーレスポンスを返します。 (O)

メッセージ ID	メッセージテキスト	メッセージの説明文
		Accept-Encoding ヘッダを確認してください。
KATR00111-E	The media type specified in the Content-Type header is unsupported. Content-Type ヘッダのメディアタイプはサポート外です。	リクエストに Content-Type ヘッダが付与されていないか、または Content-Type ヘッダで指定されているメディアタイプが Tuning Manager Server がサポートしていない形式です。 (S) 処理を中断し、エラーレスポンスを返します。 (O) Content-Type ヘッダを確認してください。
KATR00112-E	The format of the JSON object specified in the request body is invalid. リクエストボディに指定された JSON オブジェクトが不正な形式です。	次のどれかの要因に該当しています。 <ul style="list-style-type: none"> リクエストボディに指定しているオブジェクトが、サポートしていないスキーマの JSON オブジェクトです。 Content-Length を付与している場合、指定しているボディサイズとボディに指定したオブジェクトのサイズが一致しないため、JSON オブジェクトを途中までしか読み込んでいません。 Transfer-Encoding を付与している場合、ボディ内で指定しているチャンクサイズとその直後に続くチャンクのサイズが一致しないため、JSON オブジェクトを途中までしか読み込んでいません。 (S) 処理を中断し、エラーレスポンスを返します。 (O) 次の対処をしてください。 <ul style="list-style-type: none"> リクエストボディを確認してください。 Content-Length に指定した値とボディサイズを付与した JSON オブジェクトのサイズを確認してください。 指定したチャンクサイズとチャンクの実サイズを確認してください。
KATR00113-E	The request body is too long. リクエストボディの文字列長が規定サイズを超えました。	リクエストボディの文字列長が規定サイズを超えました。 (S) 処理を中断し、エラーレスポンスを返します。 (O) リクエストボディの内容を見直してください。
KATR10001-E	An unexpected error occurred during license authentication. (details =<ライセンスコマンドの戻り値>) ライセンス認証処理中に予期しないエラーが発生しました。詳細情報 : <ライセンスコマンドの戻り値>	ライセンス情報の取得に失敗しました。 (S) 処理を中断し、エラーレスポンスを返します。 (O) システム管理者に連絡してください。 問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必

メッセージID	メッセージテキスト	メッセージの説明文
		要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。
KATR10002-E	The system environment is invalid. (maintenance information = <コマンド名>, <コマンドの戻り値>) システム環境が不正です。詳細情報 : <コマンド名>, <コマンドの戻り値>	システム環境が不正です。 (S) 処理を中断し、エラーレスポンスを返します。 (O) システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。
KATR10005-E	Enter a license key or license key file. ライセンスキーまたはライセンスキーファイルを入力してください。	ライセンスキーまたはライセンスキーファイルが入力されていません。 (S) 処理を中断し、エラーレスポンスを返します。 (O) ライセンス登録してください。
KATR10006-E	No valid license exists. 有効なライセンスがありません。	ライセンスが入力されていません。 (S) 処理を中断し、エラーレスポンスを返します。 (O) Tuning Manager のサーバ管理者に連絡してください。
KATR10007-E	The temporary license has expired. Enter a product license key or a product license key file. 一時ライセンスの有効期間が満了しています。正規のライセンスキーまたはライセンスキーファイルを入力してください。	一時ライセンスキーが有効期限切れです。 (S) 処理を中断し、エラーレスポンスを返します。 (O) ライセンスを購入し、ライセンス登録してから、再実行してください。
KATR10008-E	The emergency license has expired. Enter a product license key or a product license key file. 非常ライセンスの有効期間が満了しています。正規のライセンスキーまたはライセンスキーファイルを入力してください。	非常ライセンスキー有効期限切れです。 (S) 処理を中断し、エラーレスポンスを返します。 (O) ライセンスを購入し、ライセンス登録してから、再実行してください。
KATR10009-E	All of the subscription license has expired. Enter a product license key or a product license key file. 全ての装置のライセンス期限が満了しています。正規のライセンスキーまたはライセンスキーファイルを入力してください。	すべての Subscription ライセンスキーが有効期限切れです。 (S) 処理を中断し、エラーレスポンスを返します。 (O) ライセンスを購入し、ライセンス登録してから、再実行してください。
KATR10010-E	The user ID or password is incorrect. ユーザー ID, またはパスワードが違います。	次のどれかの要因に該当しています。 ・ ユーザー ID またはパスワードの入力が不正であった場合 ・ ユーザーが操作権限を持っていない場合 ・ ユーザーがアカウントロックされている場合 (S)

メッセージ ID	メッセージテキスト	メッセージの説明文
		<p>処理を中断し、エラーレスポンスを返します。</p> <p>(O)</p> <p>次の対処をしてください。</p> <ul style="list-style-type: none"> ・ユーザー ID, パスワードを確認し、再入力してください。 ・管理者の権限を見直し、適切な権限を付加してください。 ・アカウントが有効か確認してください。
KATR10011-E	The user ID or password is incorrect. ユーザー ID, またはパスワードが違います。	<p>ユーザー ID, またはパスワードが違います。</p> <p>(S)</p> <p>処理を中断し、エラーレスポンスを返します。</p> <p>(O)</p> <p>ユーザー ID, パスワードを確認し、再入力してください。</p>
KATR10012-E	Authentication failed. 認証に失敗しました。	<p>次のどちらかの要因に該当しています。</p> <ul style="list-style-type: none"> ・ SSO サービスが起動していない場合 ・ その他、予期しない問題が発生した場合 <p>(S)</p> <p>処理を中断し、エラーレスポンスを返します。</p> <p>(O)</p> <p>しばらく待って再実行してください。再度エラーが発生する場合は、Tuning Manager Server を再起動してください。問題が解決しない場合は、システム管理者に連絡してください。それでも問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。</p>
KATR10013-E	Authentication failed. 認証に失敗しました。	<p>認証に失敗しました。</p> <p>(S)</p> <p>処理を中断し、エラーレスポンスを返します。</p> <p>(O)</p> <p>しばらく待って再実行してください。それでも直らない場合は、Tuning Manager のサーバ管理者に連絡してください。</p>
KATR10015-E	Communication with an external authentication server has failed. 外部認証サーバとの通信に失敗しました。	<p>外部認証サーバでユーザー認証したときに発生した通信エラーです。</p> <p>(S)</p> <p>処理を中断し、エラーレスポンスを返します。</p> <p>(O)</p> <p>次の対処をしてください。</p> <p>外部認証サーバが LDAP の場合：</p> <p>1.exauth.properties に設定されているホスト、ポートおよびプロトコルを使用して外部認証サーバにアクセスできるかどうかを確認してください。また、外部認証サーバの設定またはネットワークの状態を確認してください。</p>

メッセージID	メッセージテキスト	メッセージの説明文
		<p>2.hcmdsldapuser コマンドで exauth.properties の auth.server.name 属性に指定したサーバ名の情報が登録されていることを確認してください。</p> <p>3.StartTLS で通信するときは、SSL の設定を確認してください。</p> <p>4.問題が解決しないときは、システム管理者に連絡してください。それでも問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。</p> <p>外部認証サーバが RADIUS の場合：</p> <p>1.exauth.properties に設定されているホスト、ポートおよびプロトコルを使用して外部認証サーバにアクセスできるかどうかを確認してください。また、外部認証サーバの設定またはネットワークの状態を確認してください。</p> <p>2.hcmdsradiussecret コマンドで exauth.properties の auth.server.name 属性に指定したサーバ名の情報が登録されていることを確認してください。</p> <p>3.問題が解決しないときは、システム管理者に連絡してください。それでも問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。</p> <p>外部認証サーバが Kerberos の場合：</p> <p>1.exauth.properties の default_realm 属性および kdc 属性に指定した情報が正しいことを確認してください。</p> <p>2.ネットワークが正しく接続されていることを確認してください。</p> <p>3.外部認証サーバが Kerberos V5 プロトコルに対応（準拠）していることを確認してください。</p> <p>4.問題が解決しないときは、システム管理者に連絡してください。それでも問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。</p> <p>exauth.properties の詳細については、マニュアル「Hitachi Device Manager and Provisioning Manager システム構成ガイド」を参照してください。</p>
KATR10016-E	<p>A failure occurred during authentication.</p> <p>認証処理で障害が発生しています。</p>	<p>Tuning Manager のサーバまたは認証サーバで障害が発生しています。</p> <p>(S)</p> <p>処理を中断し、エラーレスポンスを返します。</p> <p>(O)</p>

メッセージ ID	メッセージテキスト	メッセージの説明文
		Tuning Manager のサーバ管理者に連絡してください。
KATR10017-E	Communication with an authentication server has failed. 認証サーバとの通信に失敗しました。	共通コンポーネントが起動していないか、接続障害が発生しています。 (S) 処理を中断し、エラーレスポンスを返します。 (O) 次の対処をしてください。 1.共通コンポーネントが起動しているかどうか確認してください。 2.通信障害が発生していないか確認してください。
KATR10018-E	The Common Component has blocked Tuning Manager from accessing the database. Contact the server administrator. Tuning Manager のサーバが使用する DB が閉塞しました。閉塞した部位は、共通コンポーネントです。サーバ管理者に連絡してください。	データベースの RD エリアが閉塞しています。 (S) 処理を中断し、エラーレスポンスを返します。 (O) システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。
KATR10019-E	The authentication parameter is invalid. 認証パラメーターが不正です。	認証方式パラメーターに不正な文字列が指定されています。 (S) 処理を中断し、エラーレスポンスを返します。 (O) パラメーターを見直し、認証し直してください。
KATR10020-E	An unexpected error occurred during user authentication. ユーザー認証処理中に予期しないエラーが発生しました。	共通コンポーネントのユーザー情報取得に失敗しました。 (S) 処理を中断し、エラーレスポンスを返します。 (O) システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。
KATR10040-E	The Authentication HTTP header, which is required for authentication, is not specified. 認証に必要な HTTP ヘッダ (Authorization ヘッダ) が指定されていません。	Authorization ヘッダが指定されていません。 (S) エラーレスポンスを返します。 (O) Authorization ヘッダに認証に必要な値を設定してください。
KATR10041-I	Tuning Manager REST Application Server will now start. Tuning Manager REST Application Server の起動処理を開始します。	Tuning Manager REST Application Server の起動処理を開始しました。

メッセージID	メッセージテキスト	メッセージの説明文
KATR10042-I	Initialization of Tuning Manager REST Application Server finished. Tuning Manager REST Application Server の初期化処理が完了しました。	Tuning Manager REST Application Server の初期化処理が完了しました。
KATR10043-I	Tuning Manager REST Application Server will now stop. Tuning Manager REST Application Server の停止処理を開始します。	Tuning Manager REST Application Server の停止処理を開始しました。
KATR10044-I	Tuning Manager REST Application Server stopped. Tuning Manager REST Application Server の終了処理が完了しました。	Tuning Manager REST Application Server の終了処理が完了しました。
KATR10045-E	The specified meta information does not exist. (specified meta-information type = <指定されたメタ情報種別>) 指定したメタ情報が存在しません。(指定メタ情報種別:<指定されたメタ情報種別>)	指定したメタ情報が存在しません。 (O) 正しいメタ情報名を指定して再実行してください。
KATR10046-E	An attempt to read an internal file has failed. (maintenance information = <プロパティファイル名>, <エラーコード>) 内部ファイルの読み込みに失敗しました。保守情報<プロパティファイル名>, <エラーコード>	内部ファイルの読み込みに失敗しました。原因は次のとおりです。 エラーコード 0 : ファイルが存在しません。 1 : ファイルへのアクセス権限がありません。 2 : ファイルの形式が不正です。 (S) 初期化処理を終了します。 (O) 事前にバックアップを実施の上, 上書きインストールしてください。
KATR10047-E	An attempt to read a property file has failed. (property file name = <プロパティファイル名>, error code = <エラーコード>) プロパティファイルの読み込みに失敗しました。(プロパティファイル名 : <プロパティファイル名>, エラーコード : <エラーコード>)	プロパティファイルの読み込みに失敗しました。原因は次のとおりです。 エラーコード 0 : ファイルが存在しません。 1 : ファイルへのアクセス権限がありません。 2 : ファイルの形式が不正です。 (S) 初期化処理を終了します。 (O) エラーコードの値に応じて次の対応を実施してください。 エラーコード 0 : 該当のプロパティファイルを適切な場所に配置してください。 1 : ファイルのアクセス権限の設定を見直してください。 2 : 正しい形式にしてください。 プロパティファイルの詳細については, マニュアル「Hitachi Command Suite Tuning Manager - Agents」を参照してください。
KATR10048-E	An unknown property has been specified. (property name = <プロパティ名>)	未知のプロパティが設定されています。 (S)

メッセージ ID	メッセージテキスト	メッセージの説明文
	未知のプロパティが設定されています。 (プロパティ名:<プロパティ名>)	ほかのプロパティのチェックを続行して、その後システムを終了します。 (O) プロパティの設定を見直してください。
KATR10049-E	The value specified for a property is invalid. (property file name =<プロパティファイル名>, property name =<プロパティ名>, specified value =<設定値>) プロパティの設定値が不正です。(プロパティファイル名:<プロパティファイル名>, プロパティ名:<プロパティ名>, 設定値:<設定値>)	プロパティの設定値が不正です。 (S) 初期化処理を終了します。 (O) プロパティの設定を見直してください。
KATR10050-E	An internal file is invalid. (maintenance information =<プロパティ名>) 内部ファイルが不正です。保守情報:<プロパティ名>	システムプロパティに未知のプロパティが設定されています。 (S) 処理を継続します。 (O) 事前にバックアップを実施の上、上書きインストールしてください。
KATR10051-E	An internal file is invalid. (maintenance information =<プロパティファイル名>, <プロパティ名>, <設定値>) 内部ファイルが不正です。保守情報:<プロパティファイル名>, <プロパティ名>, <設定値>	システムプロパティの設定値が不正です。 (S) 初期化処理を終了します。 (O) 事前にバックアップを実施の上、上書きインストールしてください。
KATR10140-W	プロパティの設定値が不正です。(プロパティファイル名:<プロパティファイル名>, プロパティ名:<プロパティ名>, 設定値:<設定値>)	プロパティの設定値が不正です。 (S) デフォルト値を設定して処理を継続します。 (O) プロパティの設定を見直してください。
KATR11002-E	PFM-Manager could not be accessed. Check the status of PFM-Manager, and then try to refresh the agent information again. PFM-Manager にアクセスできませんでした。PFM-Manager の状態を確認し、エージェント情報のリフレッシュを再度実行してください。	PFM-Manager が停止しています。 (S) エージェント情報のリフレッシュを中断します。 (O) PFM-Manager の状態を確認してください。
KATR11003-W	Detection of Tuning Manager - Agent REST Web Service on an agent host failed. (host name =<検出対象のホスト名>, protocol =<検出に利用したプロトコル>, IP address =<検出に利用した IP アドレス>, port =<検出に利用したポート>, status code =<エージェントホストから返却された HTTP ステータスコード>) エージェントホストの Tuning Manager - Agent REST Web Service の検出に失敗しました。(ホスト名:<検出対象のホスト名>, プロトコル:<検出に利用したプロトコル>, IP アドレス:	次のどれかに該当するおそれがあります。 • エージェントのバージョンが古く、Tuning Manager - Agent REST Web Service がインストールされていない。 • エージェントホストの Tuning Manager - Agent REST Web Service が起動していない。 • エージェントホストの Tuning Manager - Agent REST Web Service の設定を変更したが、Tuning Manager Server ホスト上で実行する手順を実行していない。 • エージェントホストの Tuning Manager - Agent REST Web Service との通信ができない。

メッセージID	メッセージテキスト	メッセージの説明文
	<検出に利用した IP アドレス>, ポート : <検出に利用したポート>, ステータスコード : <エージェントホストから返却された HTTP ステータスコード>	<p>(S) 処理を続行します。</p> <p>(O)</p> <ul style="list-style-type: none"> エージェントのバージョンが v8.0.0 より前の場合は, エージェントをバージョンアップしてください。 エージェントホストの Tuning Manager - Agent REST Web Service の状態を確認してください。 エージェントホストの Tuning Manager - Agent REST Web Service の設定を変更した場合は, Tuning Manager Server ホスト上で必要な手順を実行したか確認してください。 ネットワーク設定を見直してください。
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 "<監視対象の特定に失敗したエージェントインスタンスの名前>". エージェントの監視対象を特定できませんでした。エージェント(<監視対象の特定に失敗したエージェントインスタンスの名前>)で Tuning Manager API 向けのデータ管理が有効になっているか確認してください。	<p>対象エージェントの稼働性能情報ファイル出力機能が有効になっていないおそれがあります。</p> <p>(S) エージェント情報のリフレッシュを続行します。</p> <p>(O) 対象エージェントの設定・状態を確認してください。</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 "<アクセスに失敗したエージェントサーバのホスト名>". (status code =<アクセス失敗時の HTTP ステータスコード>) エージェントの Tuning Manager - Agent REST Web Service に正しくアクセスできませんでした。ホスト(<アクセスに失敗したエージェントサーバのホスト名>)の Tuning Manager - Agent REST Web Service の状態を確認してください。Status Code:<アクセス失敗時の HTTP ステータスコード>	<p>対象ホストの Tuning Manager - Agent REST Web Service が, エラー応答を返しています。</p> <p>(S) エージェント情報のリフレッシュを続行します。</p> <p>(O) 対象ホストの Tuning Manager - Agent REST Web Service の状態を確認してください。</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 "<アクセスに失敗したエージェントサーバのホスト名>". エージェントの Tuning Manager - Agent REST Web Service にアクセスできませんでした。ホスト(<アクセスに失敗したエージェントサーバのホスト名>)の Tuning Manager - Agent	<p>対象ホストの Tuning Manager - Agent REST Web Service から応答が返ってきません。</p> <p>(S) エージェント情報のリフレッシュを続行します。</p> <p>(O) 対象ホストの Tuning Manager - Agent REST Web Service の状態を確認してください。対象ホストの Tuning Manager - Agent REST Web Service で HTTPS</p>

メッセージ ID	メッセージテキスト	メッセージの説明文
	REST Web Service の状態を確認してください。	の設定をした場合、Tuning Manager Server に証明書のインポートをしているか確認してください。
KATR11007-I	Periodic refreshing of agent information will be skipped because the agent information is currently being refreshed. エージェント情報のリフレッシュ実行中のため、エージェント情報の定期リフレッシュをスキップします。	エージェント情報のリフレッシュ実行中に、定期リフレッシュ開始のタイミングになりました。
KATR11008-E	An HTTP header is invalid. (header name = <HTTP ヘッダのキー>, value = <HTTP ヘッダの値>) HTTP ヘッダが不正です。(ヘッダ名:<HTTP ヘッダのキー>, 値:<HTTP ヘッダの値>)	指定した HTTP ヘッダが不正です。 (O) 正しい HTTP ヘッダ(<HTTP ヘッダのキー>)を指定して、リトライしてください。
KATR11009-E	Specify the "<HTTP ヘッダのキー>" HTTP header correctly, and then retry the operation. 正しい HTTP ヘッダ(<HTTP ヘッダのキー>)を指定して、リトライしてください。	正しい HTTP ヘッダ (<HTTP ヘッダのキー>) を指定して、リトライしてください。
KATR11010-E	The specified agent instance does not exist. (specified agent-instance name = <指定されたインスタンス ID>) 指定した Agent インスタンスが存在しません。(指定 Agent インスタンス名:<指定されたインスタンス ID>)	指定した Agent インスタンスが存在しません。 (O) API を実行可能な Agent インスタンスを指定してリトライしてください。
KATR11011-E	Specify an agent instance that can execute API, and then retry the operation. API を実行可能な Agent インスタンスを指定してリトライしてください。	API を実行可能な Agent インスタンスを指定してリトライしてください。
KATR11012-E	The specified record ID does not exist. (record ID = <指定されたレコード ID>) 指定したレコード ID が存在しません。(指定レコード ID:<指定されたレコード ID>)	指定したレコード ID が存在しません。 (O) Agent がサポートしているレコード ID を指定してリトライしてください。
KATR11013-E	Specify a record ID supported by the agent, and then retry the operation. Agent がサポートしているレコード ID を指定してリトライしてください。	Agent がサポートしているレコード ID を指定してリトライしてください。
KATR11014-E	Required values in the query string are missing. (<クエリ文字列で指定しなければならないキー名 複数ある場合はカンマ区切りで列挙>) クエリ文字列に必要な値が指定されていません。(クエリ文字列で指定しなければならないキー名 複数ある場合はカンマ区切りで列挙)	クエリ文字列に必要な値が指定されていません。 (O) 不足している値を指定して、リトライしてください。
KATR11015-E	Specify the missing values, and then retry the operation.	不足している値を指定して、リトライしてください。

メッセージID	メッセージテキスト	メッセージの説明文
	不足している値を指定して、リトライしてください。	
KATR11016-E	Values that can not be specified in the query string are specified. (<クエリ文字列で指定した値 複数ある場合はカンマ区切りで列挙>) クエリ文字列に指定できない値が指定されています。(<クエリ文字列で指定した値 複数ある場合はカンマ区切りで列挙>)	クエリ文字列に指定できない値が指定されています。 (O) 値を確認して、修正のうえリトライしてください。
KATR11017-E	Remove the following items from the query string, and then retry the operation: <クエリ文字列で指定した値 複数ある場合はカンマ区切りで列挙> クエリ文字列から(<クエリ文字列で指定した値 複数ある場合はカンマ区切りで列挙>)の箇所を取り除き、リトライしてください。	クエリ文字列から(<クエリ文字列で指定した値 複数ある場合はカンマ区切りで列挙>)の箇所を取り除き、リトライしてください。
KATR11018-E	In the query string, a key is specified with an invalid value. (key = <クエリ文字列で指定したキー名>, value = <クエリ文字列で指定したキーに対する値>) クエリ文字列でキー名に対して不正な値が設定されています。(キー名:<クエリ文字列で指定したキー名>, 値:<クエリ文字列で指定したキーに対する値>)	クエリ文字列でキー名に対して不正な値が設定されています。 (O) 値を確認して、修正のうえリトライしてください。
KATR11019-E	Revise the value of the key, and then retry the operation. (key = <クエリ文字列で指定したキー名>, value = <クエリ文字列で指定したキーに対する値>) <クエリ文字列で指定したキー名>の値<クエリ文字列で指定したキーに対する値>を確認して、修正のうえリトライしてください。	<クエリ文字列で指定したキー名>の値<クエリ文字列で指定したキーに対する値>を確認して、修正のうえリトライしてください。
KATR11020-E	The specified time format is invalid. (key = <クエリ文字列で指定したキー名>, value = <クエリ文字列で指定したキーに対する値>) 指定した時刻のフォーマットが不正です。(キー名:<クエリ文字列で指定したキー名>, 値:<クエリ文字列で指定したキーに対する値>)	指定した時刻のフォーマットが不正です。 (O) 値を確認してください。時刻のフォーマットは YYYY-MM-DDThh:mmZ(e.g. 2013-12-03T21:52Z)です。
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 = <クエリ文字列で指定したキー名>, value = <クエリ文字列で指定したキーに対する値>) クエリ文字列で(<クエリ文字列で指定したキー名>)に指定した値(<クエリ文字列で指定したキーに対する値>)を確認してください。時刻のフォーマット	クエリ文字列で(<クエリ文字列で指定したキー名>)に指定した値(<クエリ文字列で指定したキーに対する値>)を確認してください。時刻のフォーマットは YYYY-MM-DDThh:mmZ(e.g. 2013-12-03T21:52Z)です。

メッセージ ID	メッセージテキスト	メッセージの説明文
	は YYYY-MM-DDThh:mmZ(e.g. 2013-12-03T21:52Z)です。	
KATR11022-E	A server error occurred. (error detail = <エラー内容>) サーバエラーが発生しました。(エラー内容:<エラー内容>)	サーバエラーが発生しました。 (O) システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。
KATR11023-E	Contact the system administrator. If the problem cannot be resolved, contact Support Center, who might ask you to collect maintenance information. システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。	システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。
KATR11024-E	An attempt to load information failed, because the Agent instance is currently being initialized. (host name = ホスト名, Agent type = エージェント種別, instance name = Agent インスタンス名) Agent インスタンスが初期化処理中のため、情報取得に失敗しました。(ホスト名: ホスト名, エージェント種別: エージェント種別, インスタンス名: Agent インスタンス名)	Agent インスタンスが初期化処理中のため、情報取得に失敗しました。 (O) しばらく待ってから再実行してください。それでもエラーが発生する場合は、システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。
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. しばらく待ってから再実行してください。それでもエラーが発生する場合は、システム管理者に連絡してください。問題が解決しない場合は、顧客問い合わせ窓口へ連絡してください。	初期化処理中の Agent インスタンスに対して情報取得を行った。
KATR11026-E	An attempt to read data failed. (instance name = <指定されたインスタンス ID>, record ID = <指定されたレコード ID>, data timestamp = <データ時刻>) データの読み込みに失敗しました。(インスタンス名:<指定されたインスタンス ID>,レコード ID:<指定されたレコード ID>,データ時刻:<データ時刻>)	稼働性能情報ファイルのオープンエラーです。 (S) 対象データをスキップしてデータ取得を続けます。 (O) システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口に連絡してください。
KATR11027-E	A data-file analysis error occurred. (instance name = <指定されたインス	稼働性能情報ファイルのフォーマット不正です。

メッセージID	メッセージテキスト	メッセージの説明文
	<p>タンス ID>, record ID =<指定されたレコード ID>, data timestamp =<データ時刻>)</p> <p>データの解析エラーが発生しました。(インスタンス名:<指定されたインスタンス ID>,レコード ID:<指定されたレコード ID>,データ時刻:<データ時刻>)</p>	<p>(S) 対象データをスキップしてデータ取得を継続します。</p> <p>(O) 繰り返し問題が発生する場合には、管理者もしくはカスタマーサポートに連絡してください。</p>
KATR11028-I	<p>Periodic refreshing of agent information will be disabled because the periodic-refresh interval for agent information is set to 0.</p> <p>エージェント情報の定期リフレッシュ間隔に 0 が設定されたため、エージェント情報の定期リフレッシュを無効にします。</p>	<p>user.properties の rest.discovery.agent.interval に 0 が設定されました。</p> <p>(S) エージェント情報の定期リフレッシュを抑止します。</p>
KATR11029-W	<p>An attempt to save agent information failed. (file name =<エージェント情報の保存時に作成失敗したファイル名>)</p> <p>エージェント情報の保存に失敗しました。ファイル名=<エージェント情報の保存時に作成失敗したファイル名></p>	<p>ファイルの作成・リネームに失敗しました。</p> <p>(S) 処理を続行します。</p> <p>(O) ファイル名に表示されているファイルを出力するディレクトリについて以下を確認してください。</p> <ul style="list-style-type: none"> ・ディレクトリが存在するか ・書き込み可能になっているか
KATR11030-W	<p>An attempt to save agent information failed because deletion of old agent information failed. (file name =<エージェント情報の保存時に削除に失敗したファイル名>)</p> <p>古いエージェント情報の削除に失敗したため、エージェント情報の保存に失敗しました。ファイル名=<エージェント情報の保存時に削除に失敗したファイル名></p>	<p>ファイルの削除に失敗しました。</p> <p>(S) 処理を続行します。</p> <p>(O) ファイル名に表示されているファイルを出力するディレクトリについて以下を確認してください。</p> <ul style="list-style-type: none"> ・ディレクトリが存在するか ・書き込み可能になっているか
KATR11031-W	<p>An attempt to load agent information failed. (file name = <エージェント情報の読み込み時に読み込み失敗したファイル名>)</p> <p>エージェント情報の読み込みに失敗しました。ファイル名=<エージェント情報の読み込み時に読み込み失敗したファイル名></p>	<p>ファイルの読み込みに失敗しました。</p> <p>(S) 処理を続行します。エージェント情報が欠けた状態で起動します。</p> <p>(O) ファイル名に表示されているファイルについて以下を確認してください。問題解決後、エージェントインスタンスおよび各エージェントホストの Tuning Manager - Agent REST Web Service が起動していることを確認してからエージェント情報の更新を実行してください。</p> <ul style="list-style-type: none"> ・ファイルが存在するか ・ファイルが読み込み可能になっているか
KATR11032-E	<p>The specified agent type is unsupported. (agent type =<エージェント種別>)</p>	<p>サポート外の Agent 種別が指定されました。</p> <p>(S) 処理を中断し、エラーレスポンスを返します。</p> <p>(O)</p>

メッセージ ID	メッセージテキスト	メッセージの説明文
	指定されたエージェント種別はサポートされていません。エージェント種別=<エージェント種別>	Tuning Manager Server がサポートしている Agent 種別を確認してください。
KATR11033-E	Values that cannot be specified by using the current type of Performance database string are specified. (active mode = 現在の Performance データベースの種別, key that cannot be used = クエリ文字列で指定したキーのうち, サポートしていないキー) クエリ文字列に Performance データベースの種別でサポートしていないキーが指定されています。 (Performance データベースの種別 = 現在の Performance データベースの種別, 使用できないキー = クエリ文字列で指定したキーのうち, サポートしていないキー)	Performance データベースの種別でサポートしていないリクエストキーが指定されています。 (S) 処理を中断し, エラーレスポンスを返却する。 (O) 値を確認して, 修正のうえリトライしてください。指定したキーを利用する場合は Performance データベースの種別を Hybrid Store に変更してください。
KATR11034-E	Remove the following items from the query string, and then try the operation again. (クエリ文字列で指定したキーのうち, サポートしていないキー) If you want to use a key that you specify, change the type of the Performance database. クエリ文字列から(クエリ文字列で指定したキーのうち, サポートしていないキー)の箇所を取り除き, リトライしてください。指定したキーを利用する場合は Performance データベースの種別を変更してください。	現在の Performance データベースの種別ではサポートしていないリクエストキーが指定されている。 (S) 処理を中断し, エラーレスポンスを返却する。
KATR11037-E	Required values in the query string are missing. (<クエリ文字列で指定しなければならないキー名>) クエリ文字列に必要な値が指定されていません。(<クエリ文字列で指定しなければならないキー名>)	クエリ文字列に必要な値が指定されていません。 (O) 不足している値を指定して, リトライしてください。
KATR11038-E	In the query string, a key is specified with an invalid value. (key = <クエリ文字列で指定したキー名>, value = <クエリ文字列で指定したキーに対する値>) クエリ文字列でキー名に対して不正な値が設定されています。(キー名:<クエリ文字列で指定したキー名>, 値:<クエリ文字列で指定したキーに対する値>)	クエリ文字列でキー名に対して不正な値が設定されています。 (O) 値を確認して, 修正のうえリトライしてください。
KATR12001-W	The agent host "<指定されたエージェント ID を持つエージェントが稼働するホスト名>" is experiencing a high volume of requests. Wait a while, and then retry the operation. エージェントホスト(<指定されたエージェント ID を持つエージェントが稼	対象ホストへのリクエストが集中しています。 (S) パフォーマンスデータ取得リクエストを中断します。 (O) 対象ホストへのリクエストが集中していないか確認してください。API を多

メッセージID	メッセージテキスト	メッセージの説明文
	働するホスト名>)へのリクエストが集中しています。しばらく待ってから再度アクセスしてください。	重実行している場合は、多重度を下げてください。
KATR12002-E	The agent information is invalid. Refresh the agent information, and then retry the operation. (host = <指定されたエージェント ID を持つエージェントが稼働するホスト名>) エージェント情報が不正です。エージェント情報のリフレッシュを実行し、再度アクセスしてください。対象ホスト : <指定されたエージェント ID を持つエージェントが稼働するホスト名>	内部矛盾が発生しています。 (S) パフォーマンスデータ取得リクエストを中断します。 (O) エージェント情報のリフレッシュを実行し、再度アクセスしてください。
KATR12003-E	The version of Tuning Manager - Agent REST Web Service on the target agent is invalid. (host = <指定されたエージェント ID を持つエージェントが稼働するホスト名>) 対象エージェントの Tuning Manager - Agent REST Web Service のバージョンが不正です。ホスト : <指定されたエージェント ID を持つエージェントが稼働するホスト名>	実行している API のバージョンと対象エージェントの Tuning Manager - Agent REST Web Service のバージョンの組み合わせがサポート対象外です。 (S) パフォーマンスデータ取得リクエストを中断します。 (O) 実行する API のバージョンを変更してください。実行する API が最新のバージョンの場合、エージェントをバージョンアップしてください。
KATR12004-E	Tuning Manager - Agent REST Web Service was not detected on the target agent. (host = <指定されたエージェント ID を持つエージェントが稼働するホスト名>) 対象エージェントの Tuning Manager - Agent REST Web Service が検出できていません。ホスト : <指定されたエージェント ID を持つエージェントが稼働するホスト名>	次のどれかに該当するおそれがあります。 <ul style="list-style-type: none"> エージェントのバージョンが古く、Tuning Manager - Agent REST Web Service がインストールされていない。 エージェントホスト上の Tuning Manager - Agent REST Web Service が起動していない。 Tuning Manager サーバとエージェント Web Application の通信ができていない。 (S) パフォーマンスデータ取得リクエストを中断します。 (O) 次の対処をしてください。 <ul style="list-style-type: none"> エージェントが v8.0.0 より前の場合、エージェントをバージョンアップしてください。 エージェントホスト上の Tuning Manager - Agent REST Web Service の状態を確認してください。 ネットワーク設定を確認してください。 エージェントホスト上の Tuning Manager - Agent REST Web Service のポート番号を変更した場合は、Tuning Manager Server のプロパティを見直してください。
KATR12005-E	The specified agent does not exist. (agent type = <指定されたエージェントのエージェント種別>, instance	指定されたエージェントが存在しません。 (S)

メッセージ ID	メッセージテキスト	メッセージの説明文
	name = <指定されたエージェントのインスタンス名>, host name = <指定されたエージェントのホスト名> 指定されたエージェントは存在しません。エージェント種別:<指定されたエージェントのエージェント種別>, インスタンス名:<指定されたエージェントのインスタンス名>, ホスト名:<指定されたエージェントのホスト名>	パフォーマンスデータ取得リクエストを中断します。 (O) 指定されたエージェントに対応するエージェント情報が存在するか確認してください。
KATR12006-E	A required parameter is not specified. (parameter name = <不足しているパラメーター名>) 必要なパラメーターが指定されていません。(パラメーター名:<不足しているパラメーター名>)	パフォーマンスデータ取得時に必要な GET パラメーターが指定されていません。 (S) リクエストを中断します。 (O) 必要なパラメーターを指定しているか確認してください。
KATR12007-E	Connection from Tuning Manager to Tuning Manager - Agent REST Web Service on the agent host was refused. (host = <アクセスに失敗したエージェントホストの名前>) Tuning Manager からエージェントホスト(<アクセスに失敗したエージェントホストの名前>)の Tuning Manager - Agent REST Web Service への接続が拒否されました。	エージェントホストの Tuning Manager - Agent REST Web Service でアクセス拒否設定をしているおそれがあります。 (S) リクエストを中断します。 (O) エージェントホストの Tuning Manager - Agent REST Web Service の設定を見直してください。
KATR12008-E	Tuning Manager - Agent REST Web Service on the agent host could not be accessed. (agent host = <エージェントホスト名>) エージェントホストの Tuning Manager - Agent REST Web Service にアクセスできませんでした。エージェントホスト:<エージェントホスト名>	次のどれかに該当するおそれがあります。 <ul style="list-style-type: none"> エージェントホストの Tuning Manager - Agent REST Web Service が起動していない。 エージェントホストの Tuning Manager - Agent REST Web Service の設定変更後、エージェント情報のリフレッシュを実行していない。 エージェントホストの Tuning Manager - Agent REST Web Service と通信できない。 エージェントホストと Tuning Manager server 間の SSL の設定が不正である。 (S) 処理を中断し、エラーレスポンスを返します。 (O) 次の対処をしてください。 <ul style="list-style-type: none"> エージェントホストの Tuning Manager - Agent REST Web Service の状態を確認してください。 エージェントホストの Tuning Manager - Agent REST Web Service の設定を変更した場合は、エージェント情報のリフレッシュを実行してください。 ネットワーク設定を見直してください。 Tuning Manager Agent REST API コンポーネントに SSL を設定している

メッセージID	メッセージテキスト	メッセージの説明文
		場合は、Tuning Manager server のユーザープロパティファイル (user.properties) の暗号方式の設定または HTM-Agents ホストの htnm_httpsd.conf の設定を見直してください。
KATR12009-E	The value specified for the parameter is invalid. (parameter name = パラメータ名, value = 指定された値) パラメータに指定された値が不正です。 (パラメータ名:パラメータ名, 指定された値:指定された値)	不正な値が指定されています。 (S) 処理を中断し、エラーレスポンスを返却する。 (O) 正しい値を指定してください。
KATR12010-E	Initialization failed. 初期化に失敗しました。	初期化処理中に異常が発生しています。 (S) 処理を中断します。 (O) Tuning Manager - Agent REST Web Service を再起動してください。再度エラーが発生する場合は、システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口へ連絡してください。
KATR12014-E	Initialization is incomplete. 初期化されていません。	初期化に失敗した状態で、処理を続行しようとしていました。 (S) 処理を中断します。 (O) Tuning Manager - Agent REST Web Service を再起動してください。再度エラーが発生する場合は、システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口へ連絡してください。
KATR12025-E	A property file cannot be read. (property file = <プロパティファイル>) プロパティファイルが読み込めません。 (プロパティファイル:<プロパティファイル>)	プロパティファイルが読み込めませんでした。 (S) 処理を中断します。 (O) プロパティファイルの存在、権限を確認してください。
KATR12026-E	Check whether the property file "<プロパティファイル>" exists and its permissions. プロパティファイルの存在及び権限を確認してください。(プロパティファイル:<プロパティファイル>)	プロパティファイルの存在および権限を確認してください。
KATR12027-E	A required property is not specified. (property file = <プロパティファイル>, key = <プロパティのキー>)	必須のプロパティが指定されていません。 (S) 処理を中断します。 (O)

メッセージID	メッセージテキスト	メッセージの説明文
	必要なプロパティが指定されていません。(プロパティファイル:<プロパティファイル>, キー:<プロパティのキー>)	プロパティファイルの設定を確認してください。
KATR12028-E	Check the settings in the property file. プロパティファイルの設定を確認してください。	プロパティファイルの設定を確認してください。
KATR12029-W	A system property cannot be accessed. The default value will be used. (key =<プロパティのキー>) システムプロパティにアクセスできません。デフォルト値を使用します。(キー:<プロパティのキー>)	システムプロパティが設定されていません。 (S) 処理を続行します。 (O) システムプロパティを設定してください。
KATR12030-W	An environment variable cannot be accessed. The default value will be used. (name of environment variable =<環境変数名>) 環境変数にアクセスできません。デフォルト値を使用します。(環境変数名:<環境変数名>)	環境変数が設定されていません。 (S) 処理を続行します。 (O) 環境変数を設定してください。
KATR12040-E	An interrupt occurred. 割り込みが発生しました。	割り込みが発生しました。 (S) 処理を中断します。
KATR12041-E	Timeout occurred. タイムアウトが発生しました。	タイムアウトが発生しました。 (S) 処理を中断します。
KATR12044-E	The number of threads exceeded the maximum. スレッド数が最大数を超えました。	リクエストが集中して、最大実行数を超えました。 (S) 処理を中断します。 (O) 時間をおいてリクエストを実行してください。
KATR12046-E	An error occurred during termination processing. 終了処理でエラーが発生しました。	終了処理でエラーが発生しました。 (S) 処理を続行します。 (O) Tuning Manager - Agent REST Web Service を再起動してください。再度エラーが発生する場合は、システム管理者に連絡してください。問題が解決しない場合は、原因究明と問題の解決をするために、詳細な調査が必要です。保守情報を採取し、顧客問い合わせ窓口へ連絡してください。
KATR12047-W	The value of a property cannot be converted to a numerical value. The default value will be used. (key =<プロパティのキー>) プロパティの設定値が数値に変換できません。デフォルト値を使用します。(キー:<プロパティのキー>)	数値を設定するプロパティの値が、数値に変換できません。 (S) 処理を続行します。 (O) 該当するプロパティの設定値を見直してください。
KATR12048-E	Tuning Manager - Agent REST Web Service on the agent host replied with	次のどちらかの要因に該当しています。

メッセージID	メッセージテキスト	メッセージの説明文
	an unexpected response. (agent host = <エージェントホスト名>) エージェントホストの Tuning Manager - Agent REST Web Service が予期しない応答を返しました。エージェントホスト: <エージェントホスト名>	<ul style="list-style-type: none"> • Tuning Manager - Agent REST Application Service が起動していない場合 • その他, 予期しない問題が発生した場合 <p>(S) 処理を中断し, エラーレスポンスを返します。</p> <p>(O) エージェントホストの Tuning Manager - Agent REST Application Service が起動しているかどうか確認してください。問題が解決しない場合は, 原因究明と問題の解決をするために, 詳細な調査が必要です。保守情報を採取し, 顧客問い合わせ窓口に連絡してください。</p>
KATR12049-E	The format of content specified in the configuration file is invalid. (file path = 設定ファイルのパス, row = 問題が発生した設定ファイルの行, specified content = 問題が発生した設定ファイルの文字列) 設定ファイルの記述内容の形式が不正です。(設定ファイルのパス = 設定ファイルのパス, 行 = 問題が発生した設定ファイルの行, 記述内容 = 問題が発生した設定ファイルの文字列)	<p>設定ファイルの記述内容が不正な形式です。</p> <p>(S) 処理を中断します。</p> <p>(O) 該当する行の記述内容を見直してください。</p>
KATR12050-E	The configuration file contains a section name that is specified more than once. (file path = 設定ファイルのパス, row = 重複したセクション名が見つかった行, section name = セクション名) 設定ファイル内のセクション名が重複しています。(設定ファイルのパス = 設定ファイルのパス, 行 = 重複したセクション名が見つかった行, セクション名 = セクション名)	<p>設定ファイル内のセクション名が重複しています。</p> <p>(S) 処理を中断します。</p> <p>(O) 該当するセクション名を見直してください。</p>
KATR12051-E	The configuration file contains a key name that is specified more than once. (file path = 設定ファイルのパス, row = 重複したキー名が見つかった行, key name = キー名) 設定ファイル内のキー名が重複しています。(設定ファイルのパス = 設定ファイルのパス, 行 = 重複したキー名が見つかった行, キー名 = キー名)	<p>設定ファイル内のキー名が重複しています。</p> <p>(S) 処理を中断します。</p> <p>(O) 該当するキー名を見直してください。</p>
KATR12052-E	Failed to read a properties file. (file = プロパティファイル名, section = セクション名, label = ラベル名) プロパティファイルの読み込みに失敗しました。(file: プロパティファイル名, section: セクション名, label: ラベル名)	<p>プロパティファイルの読み込みに失敗しました。</p> <p>(S) 処理を中断します。</p> <p>(O) プロパティファイルの権限を確認してください。権限がある場合は, プロパティの設定を見直してください。</p>

メッセージID	メッセージテキスト	メッセージの説明文
KATR12053-E	Performance data could not be acquired because initialization processing failed after a restoration or migration. (host name = ホスト名, agent type = エージェント種別, instance name = Agent インスタンス名) リストア後または移行後の初期化処理で失敗しているため、パフォーマンスデータが取得できません。(ホスト名: ホスト名, エージェント種別: エージェント種別, インスタンス名: Agent インスタンス名)	リストア (htmhstore コマンド) または移行 (htmhconvert コマンド, または htmhmigrate コマンド) の対象データが破損しています。 (S) パフォーマンスデータ取得リクエストを中断します。 (O) リストア処理 (htmhstore コマンド), または移行処理 (htmhconvert コマンド, または htmhmigrate コマンド) の前提条件を満たしているか確認してください。問題がある場合には、前提条件を満たした上で、マニュアルの手順に従い、再度実施してください。
KATR12054-E	The specified information cannot be loaded in Tuning Manager API. (agent type = 指定されたエージェントのエージェント種別, instance name = 指定されたエージェントのインスタンス名, host name = 指定されたエージェントのホスト名) 指定された情報を Tuning Manager API で取得することができません。(エージェント種別: 指定されたエージェントのエージェント種別, インスタンス名: 指定されたエージェントのインスタンス名, ホスト名: 指定されたエージェントのホスト名)	指定された情報を Tuning Manager API で取得することができません。 (S) パフォーマンスデータ取得リクエストを中断します。 (O) 指定されたホストに対応するエージェントが存在するか、または指定されたエージェントが Tuning Manager API に対応しているかを確認してください。
KATR15002-E	A variable that is not defined in a schema is specified. (alert definition name = アラート定義名, variable name = 変数名) スキーマに定義されていない変数が指定されています。(アラート定義名 = アラート定義名, 変数名 = 変数名)	スキーマに定義されていない変数が指定されています。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KATR15003-E	A value of an invalid type is specified. (alert definition name = アラート定義名, variable name = 変数名) 不正な型の値が設定されています。(アラート定義名 = アラート定義名, 変数名 = 変数名)	不正な型の値が設定されています。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KATR15004-E	The alert definition has already been defined. (alert definition name = アラート定義名) すでに同名のアラート定義が定義されています。(アラート定義名 = アラート定義名)	すでに同名のアラート定義が定義されています。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KATR15005-E	The array length specified for 変数名 exceeds the maximum limit. (alert definition name = アラート定義名, value = 変数に指定した値, maximum limit = 上限値)	要素数が上限を超えています。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。

メッセージID	メッセージテキスト	メッセージの説明文
	変数名に定義した要素数が上限を超えています。(アラート定義名 = アラート定義名, 値 = 変数に指定した値, 上限 = 上限値)	
KATR15006-E	An invalid value is specified for 変数名. (alert definition name = アラート定義名, value = 変数に指定した値) 変数名に不正な値が設定されています。(アラート定義名 = アラート定義名, 値 = 変数に指定した値)	不正な値が設定されています。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KATR15007-E	The string length specified for 変数名 exceeds the maximum limit. (alert definition name = アラート定義名, value = 変数に指定した値, maximum limit = 上限値) 変数名に指定した文字列長が上限を超えています。(アラート定義名 = アラート定義名, 値 = 変数に指定した値, 上限 = 上限値)	指定した文字列長が上限を超えています。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KATR15008-E	A value is not specified for the variable 変数名. (alert definition name = アラート定義名) 変数変数名に値が指定されていません。(アラート定義名 = アラート定義名)	値が指定されていません。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KATR15009-E	The JSON format is invalid. (alert definition name = アラート定義名, detailed information = エラーの詳細情報) JSON の形式が不正です。(アラート定義名 = アラート定義名, 詳細情報 = エラーの詳細情報)	JSON の形式が不正です。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KATR15010-E	The specified field does not exist. (alert definition name = アラート定義名, field name = フィールド名) 指定されたフィールドは存在しません。(アラート定義名 = アラート定義名, フィールド名 = フィールド名)	指定されたフィールドは存在しません。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KATR15012-E	The format version is invalid. (alert definition name = アラート定義名, format version = フォーマットバージョン) フォーマットのバージョンが不正です。(アラート定義名 = アラート定義名, フォーマットバージョン = フォーマットバージョン)	フォーマットのバージョンが不正です。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KATR15013-E	An unexpected internal error occurred. 予期せぬ内部エラーが発生しました。	予期せぬ内部エラーが発生しました。 (S) 処理を中断します。 (O) システム管理者に連絡してください。 問題が解決しない場合は、顧客問い合わせ窓口へ連絡してください。

メッセージ ID	メッセージテキスト	メッセージの説明文
KATR15014-E	The specified Agent instance does not exist. 指定された Agent インスタンスは存在しません。	指定された Agent インスタンスは存在しません。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KATR15015-E	The alert definitions of the Device Manager server and Tuning Manager Agent do not match. Device Manager Server と Tuning Manager Agent のアラート定義が不一致です。	Device Manager Server と Tuning Manager Agent のアラート定義が不一致です。 (S) 処理を中断します。 (O) Device Manager Server と Tuning Manager Agent のアラート定義を同期してください。
KATR15016-E	The Performance database type is not Hybrid Store. Performance データベースが Hybrid Store ではありません。	Performance データベースが Hybrid Store ではありません。 (S) 処理を中断します。 (O) Hybrid Store を使用しているエージェントを指定してください。
KATR15017-E	The alert definition names specified in the query and the request body are different. クエリで指定したアラート定義名とリクエストボディで指定したアラート定義名が異なります。	クエリで指定したアラート定義名とリクエストボディで指定したアラート定義名が異なります。 (S) 処理を中断します。 (O) リクエストボディ、クエリの内容を見直してください。
KATR15018-E	The specified alert definition does not exist. (alert definition name = アラート定義名) 存在しないアラート定義が指定されています。(アラート定義名 = アラート定義名)	存在しないアラート定義が指定されています。 (S) 処理を中断します。 (O) 指定するアラート定義名を見直してください。
KATR15019-E	Multiple records are specified for a variable for which only one record can be specified. (alert definition name = アラート定義名, variable name = レコード名) 単一のレコードしか指定できない変数に、複数のレコードが指定されています。(アラート定義名 = アラート定義名, 変数名 = レコード名)	単一のレコードしか指定できない変数に、複数のレコードが指定されています。 (S) 処理を中断します。 (O) 次の内容でリクエストボディを見直してください。 ◦ 1つの target 型内の filterCondition 型には、すべて同じレコード名を指定しているか ◦ 1つの trigger 型内の condition 型には、すべて同じレコード名を指定しているか ◦ 1つの definition 型内で指定している ID の target 型および trigger 型には、すべて同じレコード名を指定しているか

メッセージID	メッセージテキスト	メッセージの説明文
KATR15020-E	An empty string is specified for the variable 変数名. (alert definition name = アラート定義名) 変数変数名に空文字が指定されています。(アラート定義名 = アラート定義名)	空文字が指定されています。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KATR15021-E	An empty array is specified for the variable 変数名. (alert definition name = アラート定義名) 変数変数名に空配列が指定されています。(アラート定義名 = アラート定義名)	空配列が指定されています。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KATR15022-E	Unsupported field is specified. (alert definition name = アラート定義名, field name = フィールド名) 指定できないフィールドが指定されています。(アラート定義名 = アラート定義名, フィールド名 = フィールド名)	指定できないフィールドが指定されています。 (S) 処理を中断します。 (O) リクエストボディの内容を見直してください。
KAIC08857-E	入力値が不正です。	-
KAIC08858-E	定義した Target または Notice または Trigger が Definition で使用されていません。定義種別:"< defType >", ID:"< unused-id >"	-
KAIC08859-E	指定したアラート定義はすでに存在します。アラート定義名:"< alert definition name >"	-
KAIC08860-E	未知のターゲットを監視しているインスタンスを検知しました。ホスト名 = "< hostname >", インスタンス名 = "< instance name >", ターゲット名 = "< target type >"	(O) Device Manager サーバを最新のバージョンに入れ替えてください。
KAIC08861-E	範囲外の値が指定されています。変数名 = "< element >", 最小値 = "< minimum >", 最大値 = "< maximum >"	-
KAIC08863-E	不正な値が指定されています。変数名 = "< element >", 値 = "< value >"	-
KAIC08865-E	範囲外の文字列の長さが指定されています。変数名 = "< element >", 値 = "< value >", 最小値 = "< minimum-length >", 最大値 = "< max-length >"	-
KAIC08867-E	必要な要素が存在していません。変数名 = "< element >"	-
KAIC08869-E	不正な ID が指定されています。definitionId = "< definitionId >", id = "< id >"	-
KAIC08871-E	フォーマットが不正です。	-
KAIC08873-E	Tuning Manager エージェントとの通信でエラーが発生しました。http コード = "< httpcode >", ホスト名 = "<	HTM-Agents ホストと Device Manager 間の SSL の設定が不正であるおそれがあります。 (S)

メッセージ ID	メッセージテキスト	メッセージの説明文
	<code>hostname ></code> ",プロトコル=" <code>< protocol ></code> >", ポート番号=" <code>< portnumber ></code> "	処理を中断し、エラーレスポンスを返します。 (O) Tuning Manager Agent REST API コンポーネントに SSL を設定している場合は、Device Manager の <code>tuningmanager.properties</code> または HTM-Agents ホストの <code>htnm_httpsd.conf</code> の設定を見直してください。
KAIC08874-E	Tuning Manager エージェントが起動しているか確認してください。また、Device Manager から Tuning Manager エージェントが通信できるか確認してください。	-
KAIC08875-E	Tuning Manager エージェントとの通信中に、内部エラーが発生しました。 <code>http</code> コード=" <code>< httpcode ></code> ", ホスト名=" <code>< hostname ></code> ", プロトコル=" <code>< protocol ></code> ", ポート番号=" <code>< portnumber ></code> "	-
KAIC08876-E	顧客問い合わせ窓口に連絡してください。	-
KAIC08877-E	データベースアクセス時にエラーが発生しました。	-
KAIC08878-E	Device Manager サーバが起動していることを確認してください。	-
KAIC08879-E	Tuning Manager エージェントとの通信でタイムアウトが発生しました。ホスト名=" <code>< hostname ></code> ", プロトコル=" <code>< protocol ></code> ", ポート番号=" <code>< portnumber ></code> "	-
KAIC08880-E	ネットワークに障害が発生しているか、一時的に Tuning Manager のサーバの負荷が高くなっているおそれがあります。	-
KAIC08881-E	Tuning Manager エージェントのインスタンスが存在しません。ホスト名=" <code>< hostname ></code> ", エージェント種別=" <code>< agenttype ></code> ", インスタンス名=" <code>< instancename ></code> "	(O) 次の対処をしたあと、再実行してください。 ・ ストレージシステムをリフレッシュして最新の状態にしてください。 繰り返し問題が発生する場合は、Tuning Manager のサーバ管理者に連絡してください。
KAIC08882-E	Tuning Manager エージェントが Hybrid Store モードではありません。ホスト名=" <code>< hostname ></code> ", エージェント種別=" <code>< agenttype ></code> "	-
KAIC08884-E	既に同名のアラート定義が存在します。	-
KAIC08885-E	存在しないフィールド名が指定されています。	-
KAIC08886-E	Device Manager Server と Tuning Manager Agent のアラート定義が不一致です。	-

メッセージID	メッセージテキスト	メッセージの説明文
KAIC08887-E	存在しないキー名が指定されています。	-
KAIC08888-E	リクエストが不正です。詳細情報: "<detail >"	(O) 詳細情報をもとにリクエストを修正して、再実行してください。
KAIC08911-E	指定されたアラートが存在しません。	-
KAIC08912-E	アラート名を確認してください。	-
KAIC08913-E	一つの Tuning Manager エージェントインスタンスで監視できる条件数を超過しています。変数名 = "<上限を超過している要素名>", 設定値 = "<設定値>", 上限値 = "<上限値>"	(O) 対象のインスタンスの監視に利用しているアラート定義を見直し、定義を修正・削除してください。定義を見直したあと、再度実行してください。
KAIC18809-E	データベースからデータを取得できませんでした。データベースが正常に動作していません。サーバが起動しているか確認してください。サーバが起動していない場合、サーバを起動してから再度操作してください。サーバが起動している場合は、内部エラーのおそれがあります。顧客問い合わせ窓口ご連絡してください。	データベースが正常に動作していません。 (O) サーバが起動しているか確認してください。サーバが起動していない場合、サーバを起動してから再度操作してください。サーバが起動している場合は、内部エラーのおそれがあります。顧客問い合わせ窓口ご連絡してください。
KAIC18811-E	予期しないエラーが発生しました。再度実行してください。エラーが繰り返し発生する場合には、原因究明と問題の解決には、詳細な調査が必要です。障害情報を収集し、障害対応窓口ご連絡してください。	(O) 再度実行してください。エラーが繰り返し発生する場合には、原因究明と問題の解決には、詳細な調査が必要です。障害情報を収集し、障害対応窓口ご連絡してください。
KAIC18813-E	リクエストボディの内容の読み込み時にエラーが発生しました。リクエストボディの内容を確認し、再度実行してください。	(O) リクエストボディの内容を確認し、再度実行してください。
KAIC18815-E	認証に必要な HTTP ヘッダ (Authorization ヘッダ) が指定されていません。Authorization ヘッダに認証に必要な値を設定してください。	(O) Authorization ヘッダに認証に必要な値を設定してください。
KAIC18817-E	ユーザー ID、またはパスワードが違います。ユーザー ID、パスワードを確認し、再入力してください。ユーザの権限を見直し、適切な権限を付加してください。	(O) ユーザー ID、パスワードを確認し、再入力してください。ユーザの権限を見直し、適切な権限を付加してください。
KAIC18819-E	GET パラメータに不正なキーが指定されています(キー名: <キー名>)。GET パラメータに正しいキーを設定してください。	(O) GET パラメータに正しいキーを設定してください。
KAIC18821-E	GET パラメータに指定された値の長さが不正です(キー名: <キー名>, 値: <値>)。GET パラメータの値は設定可能な範囲内で指定してください。	(O) GET パラメータの値は設定可能な範囲内で指定してください。
KAIC18823-E	GET パラメータに指定された値が不正です(キー名: <キー名>, 値: <値>)。GET パラメータの値は設定可能な範囲内で指定してください。	(O) GET パラメータの値は設定可能な範囲内で指定してください。

メッセージ ID	メッセージテキスト	メッセージの説明文
KAIC18825-E	有効なライセンスがありません。有効なライセンスを登録してください。	(O) 有効なライセンスを登録してください。

Tuning Manager API の使用方法

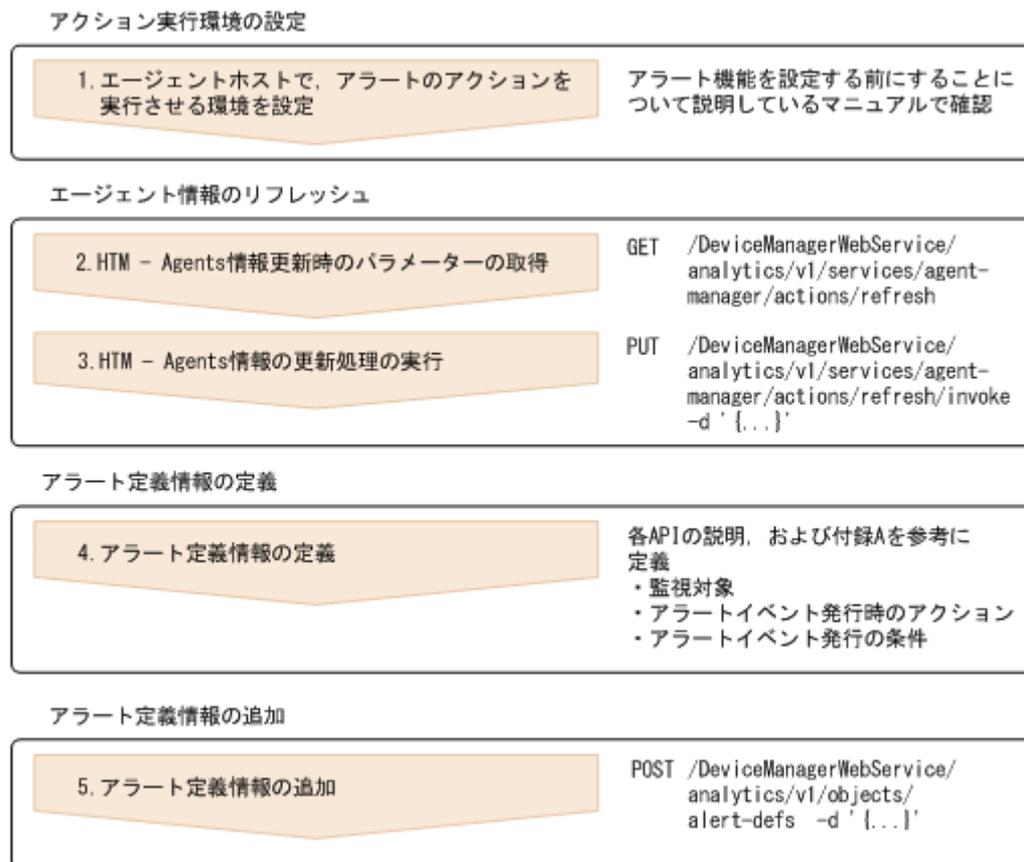
ここでは、Tuning Manager API の使用方法を説明します。

- [A.1 アラート機能を設定・運用する流れ](#)
- [A.2 アラート定義情報の作成手順](#)
- [A.3 LDEV 性能の Top5 のチャートを作成する手順](#)

A.1 アラート機能を設定・運用する流れ

アラート機能を設定・運用する流れを次の図に示します。

図 A-1 アラート機能を設定・運用する流れ



アラート機能を設定・運用する手順を次に示します。

- しきい値を超えた場合に、エージェントホストでアクションが実行される環境にします。
アクションが実行される環境にする設定手順については、マニュアル「Hitachi Command Suite Tuning Manager - Agents」のアラート機能使用時の email 送信の設定について記載している箇所を参照してください。
- Device Manager が保持しているエージェント情報をリフレッシュする際に必要となるパラメーターの情報を取得します。
- Device Manager が保持しているエージェント情報をリフレッシュします。
リフレッシュは、HTM - Agents 情報の更新を実行することで実施されます。
- アラート定義情報を定義します。
アラート定義情報には次の項目を定義します。
 - 監視対象
 - アラートイベント発行時のアクション
 - アラートイベント発行の条件
- アラート定義情報を追加します。
アラート定義情報の追加を実行してください。アラート定義情報が問題なく追加された場合、次のレスポンスが返却されます。
"state": "Succeeded"

注意

追加したアラート定義情報に誤りがないかどうかを確認するためには、アラート定義情報の参照を実行してください。誤りがあった場合には、アラート定義情報の削除を実行し、再度修正したアラート定義情報の追加を実行してください。

A.2 アラート定義情報の作成手順

ここでは、アラート定義情報の作成手順を説明します。

アラート定義情報：

アラート定義情報を次に示します。なお、見やすくするために改行を入れています。

各変数に対する値域については、オブジェクトの説明の個所を参照してください。

```
{
  "alertDefName": "<アラート定義名>",
  "targets": [{
    "targetID": "<target ID>",
    "targetDef": {
      "type": "<装置の特定>",
      "key": {
        "model": "<機種名>"
      },
      "filters": [{
        "filterConditions": [{
          "fieldName": "<フィールド>",
          "value": "<絞り込む値>"
        }]
      }]
    }
  ]},
  "notices": [{
    "noticeID": "<notice ID>",
    "emailNotification": ["warning", "caution", "normal"],
    "emails": [{
      "addresses": [
        "<emailの宛先>"
      ],
      "subject": "<emailのタイトル>",
      "body": "<emailの本文>"
    }]
  ]},
  "triggers": [{
    "triggerID": "<trigger ID>",
    "isActive": "<設定したアラート定義情報の有効/無効>",
    "monitorAlways": "<アラート機能の状態>",
    "monitoringTime": {
      "starttime": "<開始時間>",
      "endtime": "<終了時間>"
    },
    "alertEveryTime": "<状態変化時の通知有無>",
    "evaluateAllData": "<全レコードの評価方法>",
    "evaluatePoR": "<値の存在を確認するかしないか>",
    "condition": {
      "targetOfPoR": "<存在を確認する値>",
      "warning": "<異常値の条件式>",
      "caution": "<警告値の条件式>"
    },
    "frequency": {
      "count": "n/m"
    }
  ]},
  "definitions": [{
    "targetIDs": [<targetIDのリスト>],
    "noticeIDs": [<noticeIDのリスト>],
    "triggerIDs": [<triggerIDのリスト>]
  }]
}
```

A.2.1 アラート定義情報を定義する前に確認すること

アラート定義情報の定義を始める前に、定義に使用する情報および定義方法について、次の説明を読んで確認してください。

(1) アラート定義情報を適用する監視対象

アラート定義情報を適用する監視対象の定義（target 型内の targetDef 型）で使します。

ストレージシステムの場合：

- 機種名
HTM - Agent for RAID の PD レコードまたは PD_PDX レコードの DKC_NAME フィールドの値に相当します。アラート定義情報の「model」に指定する値です。
- シリアル番号※
HTM - Agent for RAID の PD レコードまたは PD_PDX レコードの DKC_SERIAL_NUMBER フィールドの値に相当します。アラート定義情報の「serial」に指定する値です。
注※ ストレージシステムをより限定したい場合にだけ、必要です。

Hitachi Virtual File Platform※の場合：

インスタンス作成時に destination_address, または detour_address で指定した IP アドレス。
アラート定義情報の「nasSystemIp」に指定する値です。

注※ 次のファイルサーバを総称して、「Hitachi Virtual File Platform」と呼びます。

- Hitachi Virtual File Platform
- Hitachi Capacity Optimization
- Hitachi Essential NAS Platform

外部 NAS Manager で管理される NAS Platform の場合：

インスタンス作成時に SMU/NAS Manager IP Address で指定した IP アドレス。
アラート定義情報の「smuIp」に指定する値です。

内部 NAS Manager で管理される NAS Platform の場合：

- NAS モジュールを搭載したストレージシステムの機種名
HTM - Agents 情報の取得の API で取得したストレージシステムの機種名を指定します。
- NAS モジュールを搭載したストレージシステムのシリアル番号
インスタンスの作成時に指定した NAS モジュールを搭載したストレージシステムのシリアル番号を指定します。

(2) アラート定義情報を適用するリソースの絞り込み条件

アラート定義情報を適用するリソースを絞り込む定義（target 型内の targetDef 型）で使します。

なお、全インスタンスに対して適用する場合、確認は不要です。

- レコード ID
- フィールド名 (PFM - Manager 名)
- 絞り込む条件値

絞り込み条件のリソース定義名の形式：

<レコード ID>.<フィールド名 (PFM - Manager 名) >

(例) PI_PRCs.ADAPTOR_ID

絞り込み条件値で使用できる文字：

- 半角英数字
- 半角スペース
- エスケープ文字※
- 特定の記号※

注※

詳細については、「A.2.2 アラート定義情報を定義する際に使用できる文字について」を参照してください。

アラート定義情報を定義する際の定義方法

定義する際の注意点：

- filters 変数内で filterCondition 型を複数定義した場合、"OR"条件で結びつきます。
- filterCondition 型内で複数定義した場合、"AND"条件で結びつきます。

リソースの絞り込み条件の定義例：

(絞り込み条件：次のどちらかに該当するリソースであること)

- PI_PRCs レコードの、Adaptor ID フィールドが A、かつ Processor ID フィールドが 1 のリソース
- PI_PRCs レコードの、Adaptor ID フィールドが B、かつ Processor ID フィールドが 2 のリソース

```
"filters":[{
  "filterConditions": [{
    "fieldName": "PI_PRCs.ADAPTOR_ID",
    "value": "A"
  },
  {
    "fieldName": "PI_PRCs.PROCESSOR_ID",
    "value": "1"
  }
]
},
{
  "filterConditions": [{
    "fieldName": "PI_PRCs.ADAPTOR_ID",
    "value": "B"
  },
  {
    "fieldName": "PI_PRCs.PROCESSOR_ID",
    "value": "2"
  }
]
}]
```

(3) アラート定義のアクション情報

アラートイベント発行時に email を送信する場合のアクション情報の定義 (notice 型内の email 型) で使用します。

- email の宛先
(例) alert@example.com
- email のタイトル

- email の本文

email のタイトルとして使用できる文字：

- 半角英数字
- 特定の記号※

注※

詳細については、「A.2.2 アラート定義情報を定義する際に使用できる文字について」を参照してください。

(例) Alert occurred in %INST%

email の本文として使用できる文字：

- 半角英数字
- 半角スペース
- エスケープ文字※
- 定義変数※
- 特定の記号※

注※

詳細については、「A.2.2 アラート定義情報を定義する際に使用できる文字について」を参照してください。

(例) %STAT%:%VAL.PI.CACHE_WRITE_PENDING_RATE% = %PROD% %NAME% on %HOST%

次に、email の例を示します。email のタイトルおよび email の本文の指定する際の参考にしてください。

図 A-2 evaluateAllData 変数に y を指定した場合の email の例

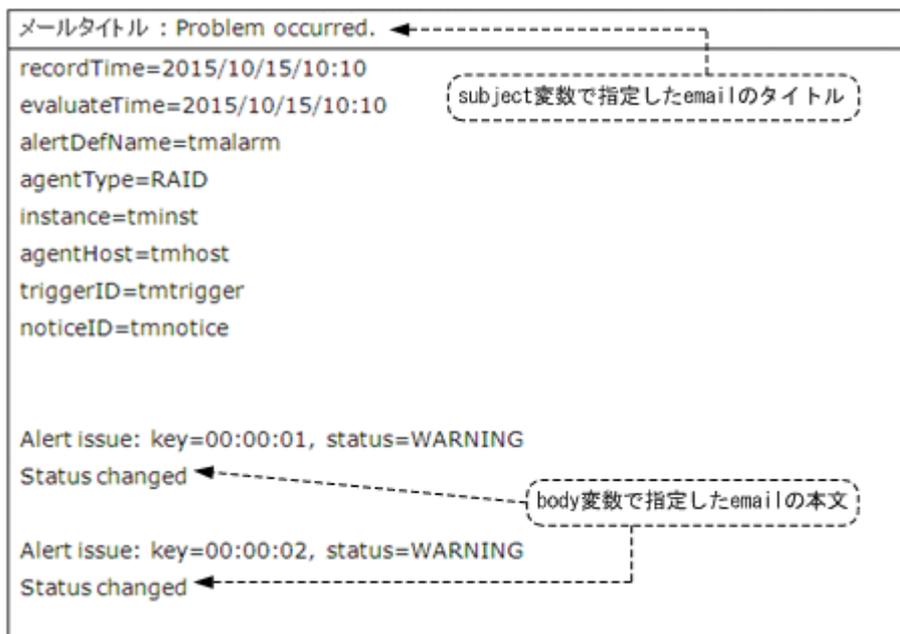
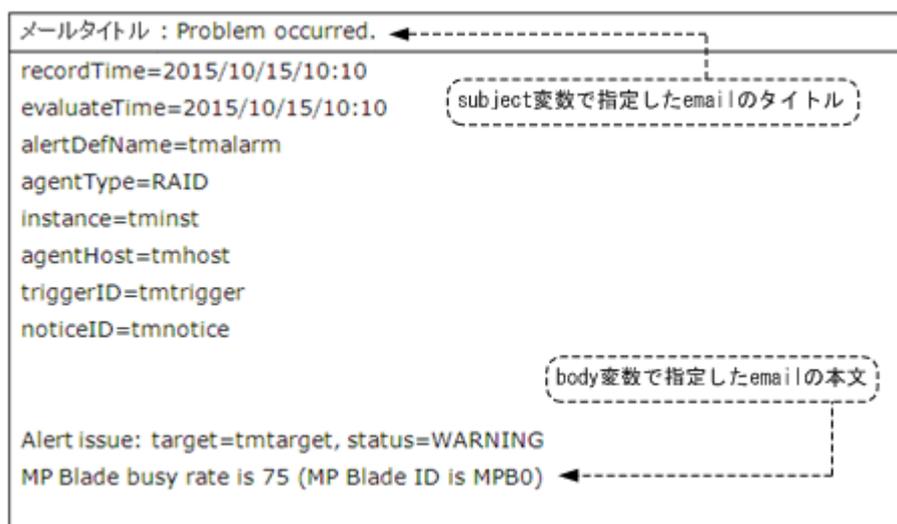


図 A-3 evaluateAllData 変数に N を指定した場合の email の例



(4) アラート定義の監視条件情報

存在を確認する値および条件式の定義（trigger 型内の condition 型）で使します。

- ・ レコード ID
- ・ フィールド名（PFM - Manager 名）

監視する方法としては、値の存在を確認して監視する方法と、異常値および警告値の条件式によって監視する方法があります。なお、次で説明する形式で使用している"△"は、半角スペースを表します。

アラート定義の監視条件情報を定義する際の定義方法（値の存在を確認して監視する場合）

定義の形式：

```
%<レコードID>.<フィールド名 (PFM - Manager 名)>%△=△'<値>'
```

<値>に使用できる文字：

- 半角英数字
- 半角スペース
- エスケープ文字※
- 特定の記号※

注※

詳細については、「A.2.2 アラート定義情報を定義する際に使用できる文字について」を参照してください。

(例) %PI.CACHE_WRITE_PENDING_RATE% = '70'

アラート定義の監視条件情報を定義する際の定義方法（異常値および警告値の条件式によって監視する場合）

異常値の条件式、および警告値の条件式は、次のどれかの形式で定義します。

注意

異常値の条件式と警告値の条件式は、条件式の右辺以外は同じにしてください。

単一条件式として定義する場合の形式：

```
%<レコードID>.<フィールド名 (PFM - Manager 名)>%△<条件式の判定記号>△'<値>'
```

- <条件式の判定記号>で指定できる値を次に示します。

表 A-1 <条件式の判定記号>で指定できる値

条件式の判定記号	説明
=	%<レコードID>.<フィールド名 (PFM - Manager 名)>%の値と<値>の値が等しい。
<>	%<レコードID>.<フィールド名 (PFM - Manager 名)>%の値と<値>の値が異なる。
>	%<レコードID>.<フィールド名 (PFM - Manager 名)>%の値が<値>の値よりも大きい。
>=	%<レコードID>.<フィールド名 (PFM - Manager 名)>%の値が<値>の値よりも大きい か、または等しい。
<	%<レコードID>.<フィールド名 (PFM - Manager 名)>%の値が<値>の値よりも小さい。
<=	%<レコードID>.<フィールド名 (PFM - Manager 名)>%の値が<値>の値よりも小さい か、または等しい。

- 絞り込み条件値で使用できる文字：

- ・半角英数字
- ・半角スペース
- ・エスケープ文字※
- ・特定の記号※

注※

詳細については、「A.2.2 アラート定義情報を定義する際に使用できる文字について」を参照してください。

(例)

「PI レコードの CACHE_WRITE_PENDING_RATE が 70%を超えた」ときに、異常値だと知らせるときの定義例

```
"warning": "%PI.CACHE_WRITE_PENDING_RATE% > '70'"
```

複合条件式として定義する場合の形式：

単一条件式の定義を、and (または AND), または or (または OR) で結びつけて定義します。

```
"<アラートイベントを発行する契機>": "<単一条件式の定義> ΔandΔ<単一条件式の定義>"
```

または

```
"<実行する契機>": "<単一条件式の定義> ΔorΔ<単一条件式の定義>"
```

(例)

PI レコードの、「CACHE_WRITE_PENDING_RATE が 70%を超えた」とき、または「CACHE_MEMORY_USAGE_RATE が 80%を超えた」ときに、異常値だと知らせるときの定義例

```
"warning": "%PI.CACHE_WRITE_PENDING_RATE% > '70' or  
%PI.CACHE_MEMORY_USAGE_RATE% > '80'"
```

数式的に順序付けて定義する場合の形式：

複合条件式を、"(") "で囲むことで、数式的に順序を付けた定義になります。

(例)

PI レコードの、「CACHE_WRITE_PENDING_RATE が 40%を超えた」かつ「CACHE_SIDE_FILE_RATE が 40%を超えた」とき、または

「CACHE_MEMORY_USAGE_RATEが80%を超えた」ときに、異常値だと知らせるときの定義例

```
"warning": "(%PI.CACHE_WRITE_PENDING_RATE% > '40' and  
%PI.CACHE_SIDE_FILE_RATE% > '40') or %PI.CACHE_MEMORY_USAGE_RATE%  
> '80'"
```

A.2.2 アラート定義情報を定義する際に使用できる文字について

ここでは、アラート定義情報を定義する際に使用できる文字について、説明します。

(1) エスケープ文字

アラート定義情報を定義する場合に使用できる次の文字を「エスケープ文字」として、説明します。

表 A-2 エスケープ文字一覧

エスケープ文字	説明
%0009	タブ
%000A	ラインフィード
%000D	改行
%0022	ダブルクォーテーション (")
%0024	\$
%0025	%
%0027	シングルクォーテーション (')
%005C	バックスラッシュ

(2) 定義変数

アラート定義情報を定義する場合に使用できる次の変数を「定義変数」として、説明します。

表 A-3 定義変数一覧

定義変数	説明
%NAME%	アラート定義名
%INST%	アラートイベントを発行したエージェントのインスタンス名
%VAL,<レコードID>.<フィールド名 (PFM・Manager名)> > %	条件式に設定したレコードの測定値
%HOST%	アラートイベントを発行したエージェントのホスト名 なお、クラスタシステムで運用している場合は、論理ホスト名になります。
%PROD%	プロダクト名
%STAT%	メッセージ出力の対象となったアラート状態
%TIME%	エージェントホストのシステム時刻 なお、通知時刻ではなく、異常が発生または通常に戻った時刻になります。

(3) 特定の記号

アラート定義情報を定義する場合に使用できる次の記号を「特定の記号」として示します。

```
! # & ( ) * - . / : ; < = > @ _ |
```

A.2.3 アラート定義情報の定義例

次にアラート定義情報の定義例を示します。

```
{
  "alertDefName": "Cache Write Pending",
  "targets": [{
    "targetID": "0",
    "targetDef": {
      "type": "model",
      "key": {
        "model": "VSP G1000"
      },
      "filters": [{
        "filterConditions": [{
          "fieldName": "PI_CLPS.CLPR_NUMBER",
          "value": "10"
        }]
      },
      {
        "filterConditions": [{
          "fieldName": "PI_CLPS.CLPR_NUMBER",
          "value": "20"
        }]
      }
    ]
  }],
  {
    "targetID": "1",
    "targetDef": {
      "type": "model",
      "key": {
        "model": "VSP G800"
      },
      "filters": [{
        "filterConditions": [{
          "fieldName": "PI_CLPS.CLPR_NUMBER",
          "value": "10"
        }]
      },
      {
        "filterConditions": [{
          "fieldName": "PI_CLPS.CLPR_NUMBER",
          "value": "20"
        }]
      }
    ]
  }],
  {
    "targetID": "2",
    "targetDef": {
      "type": "model",
      "key": {
        "model": "VSP G1000"
      }
    ]
  }],
  {
    "targetID": "3",
    "targetDef": {
      "type": "model",
      "key": {
        "model": "VSP G800"
      }
    ]
  }],
  "notices": [{
    "noticeID": "0",
    "emailNotification": ["warning", "caution"],
    "emails": [{
      "addresses": [
        "alertA@example.com"
      ]
    }
  ]
}
```

```

    ],
    "subject": "Alert occurred in %INST%",
    "body": "%STAT% = %PROD% %NAME% on %HOST%"
  },
  {
    "addresses": [
      "alertB@example.com"
    ],
    "subject": "Alert occurred in %INST%",
    "body": "%STAT% = %PROD% %NAME% on %HOST%"
  }
}],
"triggers": [
  {
    "triggerID": "0",
    "isActive": "Y",
    "monitorAlways": "N",
    "monitoringTime": {
      "starttime": "05:00",
      "endtime": "01:00"
    },
    "alertEveryTime": "N",
    "evaluateAllData": "Y",
    "evaluatePoR": "N",
    "condition": {
      "warning": "%PI_CLPS.CACHE_WRITE_PENDING_RATE% > '60'",
      "caution": "%PI_CLPS.CACHE_WRITE_PENDING_RATE% > '30'"
    },
    "frequency": {
      "count": "1/1"
    }
  },
  {
    "triggerID": "1",
    "isActive": "Y",
    "monitorAlways": "N",
    "monitoringTime": {
      "starttime": "05:00",
      "endtime": "01:00"
    },
    "alertEveryTime": "N",
    "evaluateAllData": "Y",
    "evaluatePoR": "N",
    "condition": {
      "warning": "%PI_PRCS.PROCESSOR_TYPE% > 'MPB' and
        %PI_PRCS.PROCESSOR_BUSY_RATE% >= '80'",
      "caution": "%PI_PRCS.PROCESSOR_TYPE% > 'MPB' and
        %PI_PRCS.PROCESSOR_BUSY_RATE% >= '40'"
    },
    "frequency": {
      "count": "1/1"
    }
  },
  {
    "triggerID": "2",
    "isActive": "Y",
    "monitorAlways": "N",
    "monitoringTime": {
      "starttime": "05:00",
      "endtime": "01:00"
    },
    "alertEveryTime": "N",
    "evaluateAllData": "Y",
    "evaluatePoR": "N",
    "condition": {
      "warning": "%PI_RGS.BUSY_RATE% >= '50'",
      "caution": "%PI_RGS.BUSY_RATE% >= '50'"
    },
    "frequency": {
      "count": "1/1"
    }
  }
],
{

```

```

        "triggerID": "3",
        "isActive": "Y",
        "monitorAlways": "N",
        "monitoringTime": {
            "starttime": "05:00",
            "endtime": "01:00"
        },
        "alertEveryTime": "N",
        "evaluateAllData": "Y",
        "evaluatePoR": "N",
        "condition": {
            "warning": "%PI_RGS.READ_HIT_RATE% <= '25' and
                %PI_RGS.READ_HIT_IO_COUNT% > '0'",
            "caution": "%PI_RGS.READ_HIT_RATE% <= '25' and
                %PI_RGS.READ_HIT_IO_COUNT% > '0'"
        },
        "frequency": {
            "count": "1/1"
        }
    }],
    "definitions": [{
        "targetIDs": [0,1],
        "noticeIDs": [0],
        "triggerIDs": [0]
    },
    {
        "targetIDs": [2,3],
        "noticeIDs": [0],
        "triggerIDs": [1]
    },
    {
        "targetIDs": [2,3],
        "noticeIDs": [0],
        "triggerIDs": [2,3]
    }
    ]
}

```

A.3 LDEV 性能の Top5 のチャートを作成する手順

LDEV 性能の Top5 のチャートを作成するまでの手順を説明します。

ここでは、LDEV 性能を、ストレージシステムの LDEV ごとの「Write Transfer Rate（書き込み処理の転送速度）の平均値」で捉えることとし、その平均値が大きい LDEV の上位 5 個を Top5 とします。

データ取得からチャート作成までの流れ

LDEV 性能の Top5 のチャートを作成するまでの流れを次の図に示します。

図 A-4 LDEV 性能の Top5 のチャートを作成するまでの流れ

HTM - Agents一覧のリフレッシュ

1. HTM - Agents一覧の手動リフレッシュ実行時のパラメーターの取得	GET	/TuningManager/v1/services/AgentManager/actions/refresh
2. HTM - Agents一覧の手動リフレッシュ処理の実行	POST	/TuningManager/v1/services/AgentManager/actions/refresh/invoke -d '[...]'

I/O量の多いLDEVの特定

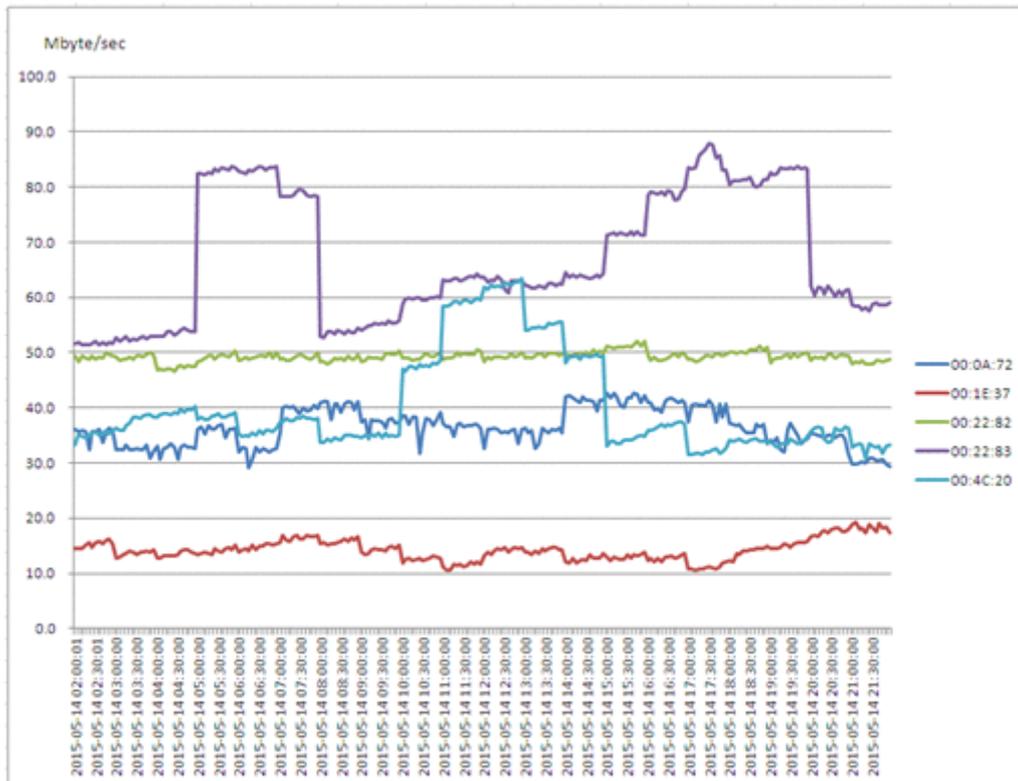
3. HTM - Agent for RAIDのインスタンス一覧の取得	GET	/TuningManager/v1/objects/AgentForRAID
4. ストレージシステムを特定し、インスタンス名およびエージェントのホスト名を確認	手順3の結果で確認	
5. 取得するレコードおよびフィールドの確認	レコードについて説明しているマニュアルで確認	
6. 時単位のパフォーマンスデータの取得	GET	/TuningManager/v1/objects/RAID_<レコードID>?...
7. 取得データを取り込み、Top5のLDEV番号の特定	表計算ソフトで確認	

特定したLDEVのチャート作成

8. Top5のLDEV番号について、分単位のパフォーマンスデータの取得	GET	/TuningManager/v1/objects/RAID_<レコードID>?...
9. 取得データを取り込み、チャートを作成	表計算ソフトで作成	

作成したチャートのイメージ図を次に示します。

図 A-5 LDEV 性能の Top5 のチャート（イメージ図）



実行例の使用ソフトウェアおよび実行環境

ここで説明する手順中の実行例は、次のソフトウェアおよび実行環境で実行した例です。

- API の実行：フリーソフトである curl コマンド（バージョン 7.19.7, Linux 版）を使用して実行
- チャート作成：Microsoft Office Excel を使用して作成
- Tuning Manager シリーズのバージョン：v8.2.0
- API クライアントがアクセスする Tuning Manager server のホスト名：server_1
- Tuning Manager server のポート番号：22015

なお、curl コマンドの実行例は見やすくするために改行を入れています。

データ取得からチャート作成までの手順および実行例

パフォーマンスデータを取得してから LDEV 性能の Top5 のチャートを作成するまでの手順および実行例を、次に示します。

1. HTM・Agents 一覧を手動リフレッシュする際に必要となるパラメーターの情報を取得します。

リクエストメッセージ

```
curl -v -u system:manager -X
GET http://server_1:22015/TuningManager/v1/services/AgentManager/
actions/refresh
```

レスポンスボディ

```
{
  "name": "RefreshAgentInstancesParameters",
  "url": "http://server_1:22015/TuningManager/v1/services/
AgentManager/actions/refresh/invoke",
  "method": "POST",
  "type": "application/json",
```

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

- 手順1.で取得したレスポンスボディを-d オプションの後ろにリクエストボディとして指定して、HTM - Agents 一覧を手動リフレッシュします。

リクエストメッセージ

```
curl -v -u system:manager -H "Content-type:application/json" -H
"Content-Length:223" -X
POST http://server_1:22015/TuningManager/v1/services/AgentManager/
actions/refresh/invoke -d
'{"name":"RefreshAgentInstancesParameters",
"url":"http://server_1:22015/TuningManager/v1/services/AgentManager/
actions/refresh/invoke",
"method":"POST","type":"application/json",
"parameters":[{"type":"string","value":"All"}]}'
```

レスポンスボディ

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

- HTM - Agent for RAID のインスタンスの情報の一覧を取得します。

リクエストメッセージ

```
curl -v -u system:manager -X GET http://server_1:22015/
TuningManager/v1/objects/AgentForRAID
```

レスポンスボディ

```
{
  "items":[{"
    "instanceName":"instance_AMS2300",
    "hostName":"agent_1",
    "ipAddr":"<agent_1 の IP アドレス>",
    "dataModelVersion":9.0,
    "storageVendor":"HITACHI",
    "storageModel":"AMS2300",
    "storageSerialNumber":"85011487",
    "lastUpdateTime":"2015-05-14T05:55:05Z"
  },
  {
    "instanceName":"instance_RAID800_6",
    "hostName":"agent_1",
    "ipAddr":"<agent_1 の IP アドレス>",
    "dataModelVersion":9.0,
    "storageVendor":"HITACHI",
    "storageModel":"VSP G1000",
    "storageSerialNumber":"10182",
    "lastUpdateTime":"2015-05-14T05:55:05Z"
  }
]}
}
```

- 手順3.のレスポンスから、チャートを作成したいストレージシステムを特定し、インスタンス名およびエージェントのホスト名を確認します。

ここでは、チャートを作成したいストレージシステムのシリアル番号が「10182」として、手順の説明を進めます。

手順3.のレスポンスから、ストレージシステムのシリアル番号を表す

"storageSerialNumber"の値が「10182」となっているアイテムを探します。このアイテムの中の次の値で、インスタンス名およびエージェントのホスト名を確認できます。

- インスタンス名を表す"instanceName"の値 : instance_RAID800_6
- エージェントのホスト名を表す"hostName"の値 : agent_1

- Tuning Manager API で取得するレコードおよびフィールドを確認します。

マニュアル「Hitachi Command Suite Tuning Manager - Agents」を参照し、Tuning Manager API で取得するレコードおよびフィールドを確認します。

ここでは、「Write Transfer Rate (書き込み処理の転送速度) の平均値」が大きい LDEV の特定を目的とするため、LDEV の性能情報に関するレコードは「PI_LDS」レコードであること、

PI_LDS レコード内の「Write Transfer Rate (書き込み処理の転送速度)」に関するフィールドは「WRITE_XFER_RATE」であることを確認します。

- 手順 4. で確認したインスタンス名とエージェントのホスト名、および手順 5. で確認したレコードとフィールドをクエリ指定し、時単位のパフォーマンスデータを取得します。

短時間で結果を得るために、時単位でデータ取得をして、扱うデータ量を削減します。

ここでは、日本時間 (UTC+9 時間) で分析期間の開始時間を「2015/5/14 2:00」、終了時間を「2015/5/14 22:00」として説明を進めます。

注意

Tuning Manager API で指定する時間は UTC 時刻です。ローカルタイムを UTC 時刻に変換してクエリ指定してください。

リクエストメッセージ

```
curl -v -u system:manager -X
GET http://server_1:22015/TuningManager/v1/objects/RAID_PI_LDS?
hostName=agent_1%26
agentInstanceName=instance_RAID800_6%26
startTime=2015-05-13T17:00Z%26
endTime=2015-05-14T13:00Z%26
fields=RECORD_TIME%1FLDEV_NUMBER%1FWRITE_XFER_RATE%26
granularity=hour
```

レスポンスボディ

```
RECORD_TIME,LDEV_NUMBER,WRITE_XFER_RATE
time_t,string(16),float
2015-05-13 17:00:01,"00:00:04",0.000000E+00
2015-05-13 17:00:01,"00:22:82",4.924448E+01
2015-05-13 17:00:01,"00:22:83",5.164448E+01
2015-05-13 17:00:01,"00:22:84",0.000000E+00
:
2015-05-13 18:00:02,"00:00:04",0.000000E+00
2015-05-13 18:00:02,"00:22:82",4.920086E+01
2015-05-13 18:00:02,"00:22:83",5.200086E+01
2015-05-13 18:00:02,"00:22:84",0.000000E+00
:
2015-05-13 19:00:02,"00:00:04",0.000000E+00
2015-05-13 19:00:02,"00:22:82",4.728886E+01
2015-05-13 19:00:02,"00:22:83",5.368886E+01
2015-05-13 19:00:02,"00:22:84",0.000000E+00
:
2015-05-14 10:00:03,"00:00:04",0.000000E+00
2015-05-14 10:00:03,"00:22:82",4.924448E+01
2015-05-14 10:00:03,"00:22:83",8.314448E+01
2015-05-14 10:00:03,"00:22:84",0.000000E+00
:
2015-05-14 11:00:01,"00:00:04",0.000000E+00
2015-05-14 11:00:01,"00:22:82",4.920086E+01
2015-05-14 11:00:01,"00:22:83",6.110086E+01
2015-05-14 11:00:01,"00:22:84",0.000000E+00
:
2015-05-14 12:00:02,"00:00:04",0.000000E+00
2015-05-14 12:00:02,"00:22:82",4.828886E+01
2015-05-14 12:00:02,"00:22:83",5.848886E+01
2015-05-14 12:00:02,"00:22:84",0.000000E+00
:
```

- 手順 6. で取得したデータを Microsoft Office Excel に取り込み、ピボットテーブル機能を使用してテーブルを作成し、目的に沿った値でソートし、Top5 を特定します。

ここでは、分析期間内の LDEV ごとの「WRITE_XFER_RATE」の平均値でソートします。

図 A-6 LDEV 性能の Top5 の LDEV 番号の特定

平均/WRITE_XFER_RATE	RECORD_TIME	2015/5/13 17:00	2015/5/13 18:00	2015/5/13 19:00	2015/5/13 20:00	2015/5/14 12:00	平均
00:22:83	51.6	52	53.6	82.9	58.4	67.1	
00:22:82	49.2	49.2	47.2	49.2	48.2	49.2	
00:4C:20	35.1	37.5	39.2	38.4	32.8	40.2	
00:0A:72	35.4	32.4	32.5	36.1	30.1	36.5	
00:1E:37	15.3	13.6	13.5	14	18.2	14.2	
00:1C:0C	10.1	10.3	14.3	13.3	10.3	10.9	
00:1E:04	1.1	2.4	3	4	2.9	3.3	
00:1C:07	6	7.1	2.9	2.8	2	3.1	
00:0B:20	1.3	1.4	2.5	1.9	1.3	1.6	
00:21:0E	0.9	1.2	1.6	0.7	1.3	1.4	
:	:	:	:	:	:	:	
00:34:62	0	0	0	0	0	0	
平均	0.036544261	0.0367394	0.037307078	0.038339111	0.036455561	0	

この結果から、分析期間内での LDEV 性能の Top5 の LDEV 番号が「00:22:83」「00:22:82」「00:4C:20」「00:0A:72」「00:1E:37」であることが特定できます。

- 手順 7. で特定した LDEV 番号について、手順 5. で確認したレコードとフィールドをクエリ指定し、分単位のパフォーマンスデータを取得します。

リクエストメッセージ

```
curl -v -u system:manager -X
GET http://server_1:22015/TuningManager/v1/objects/RAID_PI_LDS?
hostName=agent_1%26
agentInstanceName=instance_RAID800_6%26
startTime=2015-05-13T17:00Z%26
endTime=2015-05-14T13:00Z%26
fields=RECORD_TIME%1FLDEV_NUMBER%1FWRITE_XFER_RATE%26
granularity=raw%26
LDEV_NUMBER=00:22:83%1F00:22:82%1F00:4C:20%1F00:0A:72%1F00:1E:37
```

レスポンスボディ

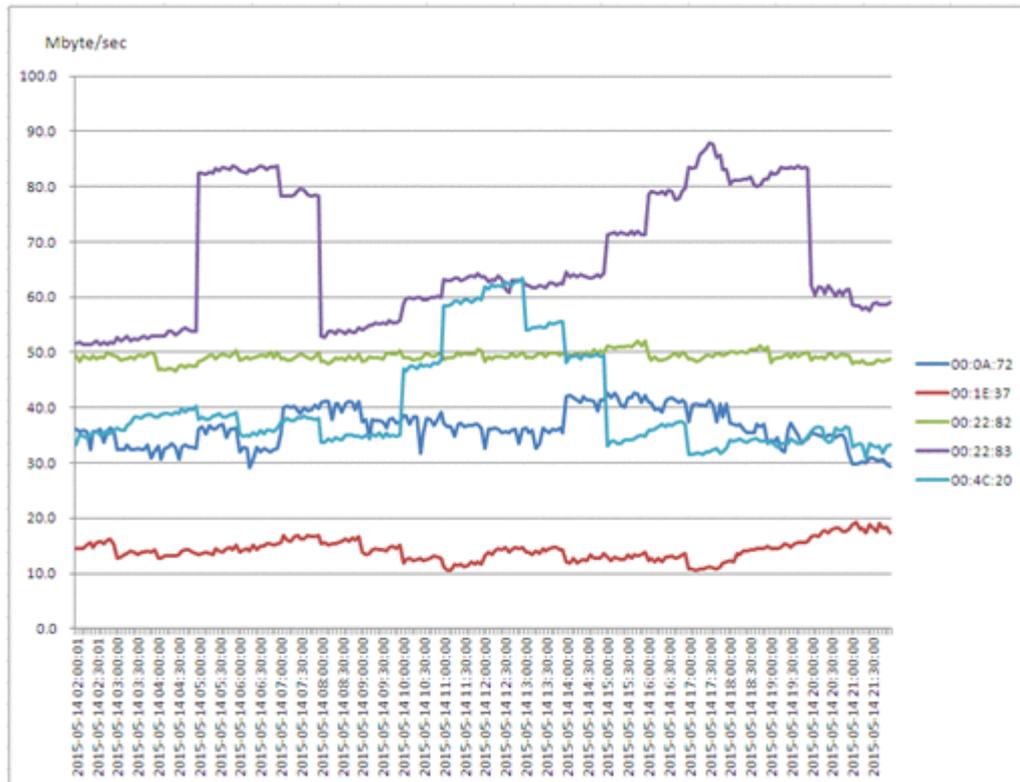
```
RECORD_TIME,LDEV_NUMBER,WRITE_XFER_RATE
time_t,string(16),float
2015-05-13 17:00:01,"00:22:83",51.656499
2015-05-13 17:05:01,"00:22:83",51.832951
2015-05-13 17:10:01,"00:22:83",51.492681
2015-05-13 17:15:01,"00:22:83",51.477599
2015-05-13 17:20:01,"00:22:83",51.488274
2015-05-13 17:25:01,"00:22:83",51.728056
2015-05-13 17:30:01,"00:22:83",52.118550
2015-05-13 17:35:01,"00:22:83",51.475923
2015-05-13 17:40:01,"00:22:83",51.957269
2015-05-13 17:45:01,"00:22:83",51.432319
2015-05-13 17:50:01,"00:22:83",51.907659
2015-05-13 17:55:01,"00:22:83",51.633032
2015-05-13 17:00:00,"00:22:83",52.657450
2015-05-13 17:05:00,"00:22:83",52.149790
2015-05-13 17:10:00,"00:22:83",52.411909
:
2015-05-14 12:45:00,"00:22:83",58.651501
2015-05-14 12:50:00,"00:22:83",58.680509
2015-05-14 12:55:00,"00:22:83",59.090207
2015-05-13 17:00:01,"00:22:82",49.514142
2015-05-13 17:05:01,"00:22:82",48.414239
2015-05-13 17:10:01,"00:22:82",49.407611
:
2015-05-14 12:20:00,"00:1E:37",17.400967
2015-05-14 12:25:00,"00:1E:37",18.788741
2015-05-14 12:30:00,"00:1E:37",18.243943
2015-05-14 12:35:00,"00:1E:37",17.527061
2015-05-14 12:40:00,"00:1E:37",19.080748
2015-05-14 12:45:00,"00:1E:37",18.271076
2015-05-14 12:50:00,"00:1E:37",18.540560
2015-05-14 12:55:00,"00:1E:37",17.407258
```

9. 手順 8.で取得したデータを Microsoft Office Excel に取り込み、グラフの機能を使ってチャートを作成します。

注意

手順 8 で取得したデータは UTC 時刻です。UTC 時刻をローカルタイムに変換してください。

図 A-7 LDEV 性能の Top5 のチャート



アラート定義情報のサンプル

ここでは、監視対象別に、アラート定義情報の JSON のサンプルを記載します。

必要に応じて、アラート定義情報を定義する際に利用してください。

□ B.1 監視対象がストレージシステムの場合

□ B.2 監視対象が NAS Platform の場合

B.1 監視対象がストレージシステムの場合

次の JSON のサンプルを使用することで、アラート機能でも、ヘルスチェックのようなストレージシステムの定期的な性能分析ができます。可変値の個所（次のサンプルの斜体の個所）は、使用する環境に応じて変更して定義してください。

なお、ストレージ種別ではなくストレージシステムごとに分析方法を変更したい場合は、`type` 変数を `block` として、`serial` 変数にストレージシステムのシリアル番号を設定してください。

B.1.1 VSP Gx00 モデル, VSP Fx00 モデル, VSP E990, HUS VM, VSP 5000 シリーズ, VSP G1000, G1500, VSP F1500, および Virtual Storage Platform シリーズの場合

```
{
  "alertDefName": "HealthCheckAlert",
  "targets": [
    {
      "targetID": "HealthCheckTarget",
      "targetDef": [
        {
          "type": "model",
          "key": {
            "model": "VSP G1000"
          }
        }
      ]
    }
  ],
  "notices": [
    {
      "noticeID": "HealthCheckNotice",
      "emails": [
        {
          "addresses": [
            "test@example.com"
          ],
          "subject": "HealthCheck Alert Result",
          "body": "Health Check Status is changed."
        }
      ]
    }
  ],
  "triggers": [
    {
      "triggerID": "CacheUtilizationTrigger",
      "condition": {
        "warning": "%PI.CACHE_WRITE_PENDING_RATE% > '70'",
        "caution": "%PI.CACHE_WRITE_PENDING_RATE% > '30'"
      }
    },
    {
      "triggerID": "CLPRCacheUtilizationTrigger",
      "condition": {
        "warning": "%PI_CLPS.CACHE_WRITE_PENDING_RATE% > '70'",
        "caution": "%PI_CLPS.CACHE_WRITE_PENDING_RATE% > '30'"
      }
    },
    {
      "triggerID": "CLPRUtilizationByMPBMPUTrigger",
      "condition": {
        "warning": "%PI_CLMS.MAX_CACHE_WRITE_PENDING_RATE% > '70'",
        "caution": "%PI_CLMS.MAX_CACHE_WRITE_PENDING_RATE% > '30'"
      }
    },
    {
      "triggerID": "MPUtilizationByMPBMPUTrigger",
      "condition": {
        "warning": "%PI_PRCS.PROCESSOR_BUSY_RATE% >= '80'",
        "caution": "%PI_PRCS.PROCESSOR_BUSY_RATE% >= '40'"
      }
    },
    {
      "triggerID": "PGUtilization",

```

```

        "condition": {
            "warning": "%PI_RGS.MAX_BUSY_RATE% >= '80'",
            "caution": "%PI_RGS.MAX_BUSY_RATE% >= '40'"
        }
    },
    "definitions": [{
        "targetIDs": ["HealthCheckTarget"],
        "noticeIDs": ["HealthCheckNotice"],
        "triggerIDs": ["CacheUtilizationTrigger",
"CLPRCacheUtilizationTrigger", "CLPRUtilizationByMPBMPUTrigger",
"MPUtilizationByMPBMPUTrigger", "PGUtilization"]
    }
]
}

```

B.1.2 VSP Gx00 モデル, VSP Fx00 モデル, VSP E990, HUS VM, VSP 5000 シリーズ, VSP G1000, G1500, VSP F1500, および Virtual Storage Platform シリーズの場合 (TCP/IP 接続を使用して収集するとき)

```

{
    "alertDefName": "HealthCheckAlert_TCPIP",
    "targets": [
        {
            "targetID": "HealthCheckTarget_TCPIP",
            "targetDef": [
                {
                    "type": "model",
                    "key": {
                        "model": "VSP G1000"
                    }
                }
            ]
        }
    ],
    "notices": [
        {
            "noticeID": "HealthCheckNotice_TCPIP",
            "emails": [
                {
                    "addresses": [
                        "test@example.com"
                    ],
                    "subject": "HealthCheck Alert Result",
                    "body": "Health Check Status is changed."
                }
            ]
        }
    ],
    "triggers": [
        {
            "triggerID": "CacheUtilizationTrigger_TCPIP",
            "condition": {
                "warning": "%PI_SCS.CACHE_WRITE_PENDING_RATE% > '70'",
                "caution": "%PI_SCS.CACHE_WRITE_PENDING_RATE% > '30'"
            }
        },
        {
            "triggerID": "CLPRCacheUtilizationTrigger_TCPIP",
            "condition": {
                "warning": "%PI_CLMX.CACHE_WRITE_PENDING_RATE% > '70'",
                "caution": "%PI_CLMX.CACHE_WRITE_PENDING_RATE% > '30'"
            }
        },
        {
            "triggerID": "CLPRUtilizationByMPBMPUTrigger_TCPIP",
            "condition": {
                "warning": "%PI_CLMX.MAX_CACHE_WRITE_PENDING_RATE% > '70'",
                "caution": "%PI_CLMX.MAX_CACHE_WRITE_PENDING_RATE% > '30'"
            }
        },
        {
            "triggerID": "MPUtilizationByMPBMPUTrigger_TCPIP",
            "condition": {
                "warning": "%PI_MPS.BUSY_RATE% >= '80'",

```

```

        "caution": "%PI_MPS.BUSY_RATE% >= '40'"
    },
    {
        "triggerID": "PGUtilization_TCPIP",
        "condition": {
            "warning": "%PI_RGU.MAX_BUSY_RATE% >= '80'",
            "caution": "%PI_RGU.MAX_BUSY_RATE% >= '40'"
        }
    }
],
"definitions": [{
    "targetIDs": ["HealthCheckTarget_TCPIP"],
    "noticeIDs": ["HealthCheckNotice_TCPIP"],
    "triggerIDs": ["CacheUtilizationTrigger_TCPIP",
"CLPRCacheUtilizationTrigger_TCPIP", "CLPRUtilizationByMPBMPUTrigger_TCPIP",
"MPUtilizationByMPBMPUTrigger_TCPIP", "PGUtilization_TCPIP"]
}
]
}

```

B.1.3 Universal Storage Platform V/VM シリーズ, Hitachi USP, および SANRISE H シリーズの場合

```

{
  "alertDefName": "HealthCheckAlert",
  "targets": [
    {
      "targetID": "HealthCheckTarget",
      "targetDef": [
        {
          "type": "model",
          "key": {
            "model": "USP_V"
          }
        }
      ]
    }
  ],
  "notices": [
    {
      "noticeID": "HealthCheckNotice",
      "emails": [
        {
          "addresses": [
            "test@example.com"
          ],
          "subject": "HealthCheck Alert Result",
          "body": "Health Check Status is changed."
        }
      ]
    }
  ],
  "triggers": [
    {
      "triggerID": "CLPRCacheUtilizationWritePendingRateTrigger",
      "condition": {
        "warning": "%PI_CLPS.CACHE_WRITE_PENDING_RATE% > '70'",
        "caution": "%PI_CLPS.CACHE_WRITE_PENDING_RATE% > '30'"
      }
    },
    {
      "triggerID": "CLPRCacheUtilizationSideFileUsageRateTrigger",
      "condition": {
        "warning": "%PI_CLPS.CACHE_SIDE_FILE_RATE% >= '20'",
        "caution": "%PI_CLPS.CACHE_SIDE_FILE_RATE% >= '10'"
      }
    },
    {
      "triggerID": "CHPUtilizationByCHATrigger",
      "condition": {
        "warning": "(%PI_PRCES.PROCESSOR_BUSY_RATE% >= '80') AND (%PI_PRCES.PROCESSOR_TYPE% = 'CHP')",
        "caution": "(%PI_PRCES.PROCESSOR_BUSY_RATE% >= '40') AND (%PI_PRCES.PROCESSOR_TYPE% = 'CHP')"
      }
    }
  ]
}

```

```

    },
    {
      "triggerID": "PGUtilizationTrigger",
      "condition": {
        "warning": "%PI_RGS.MAX_BUSY_RATE% >= '80'",
        "caution": "%PI_RGS.MAX_BUSY_RATE% >= '40'"
      }
    },
    {
      "triggerID": "DKPUtilizationByDKATrigger",
      "condition": {
        "warning": "(%PI_PRCES.PROCESSOR_BUSY_RATE% >= '80') AND
(%PI_PRCES.PROCESSOR_TYPE% = 'DKP' )",
        "caution": "(%PI_PRCES.PROCESSOR_BUSY_RATE% >= '40') AND
(%PI_PRCES.PROCESSOR_TYPE% = 'DKP' )"
      }
    }
  ],
  "definitions": [{
    "targetIDs": ["HealthCheckTarget"],
    "noticeIDs": ["HealthCheckNotice"],
    "triggerIDs": ["CLPRCacheUtilizationWritePendingRateTrigger",
"CLPRCacheUtilizationSideFileUsageRateTrigger", "CHPUtilizationByCHATrigger",
"PGUtilizationTrigger", "DKPUtilizationByDKATrigger"]
  }
]
}

```

B.1.4 HUS100 シリーズ, および Hitachi AMS2000/AMS/WMS/SMS シリーズの場合

```

{
  "alertDefName": "HealthCheckAlert",
  "targets": [
    {
      "targetID": "HealthCheckTarget",
      "targetDef": [
        {
          "type": "model",
          "key": {
            "model": "AMS2300"
          }
        }
      ]
    }
  ],
  "notices": [
    {
      "noticeID": "HealthCheckNotice",
      "emails": [
        {
          "addresses": [
            "test@example.com"
          ],
          "subject": "HealthCheck Alert Result",
          "body": "Health Check Status is changed."
        }
      ]
    }
  ],
  "triggers": [
    {
      "triggerID": "ProcessorUtilizationTrigger",
      "condition": {
        "warning": "%PI_PRCES.PROCESSOR_BUSY_RATE% >= '90'",
        "caution": "%PI_PRCES.PROCESSOR_BUSY_RATE% >= '80'"
      }
    },
    {
      "triggerID": "CacheUtilizationOnControllerTrigger",
      "condition": {
        "warning": "%PI_CLCS.CACHE_WRITE_PENDING_RATE% > '50'",
        "caution": "%PI_CLCS.CACHE_WRITE_PENDING_RATE% > '25'"
      }
    }
  ]
}

```

```

    ],
    "definitions": [{
        "targetIDs": ["HealthCheckTarget"],
        "noticeIDs": ["HealthCheckNotice"],
        "triggerIDs": ["ProcessorUtilizationTrigger",
"CacheUtilizationOnControllerTrigger"]
    }
]
}

```

B.2 監視対象が NAS Platform の場合

次の JSON のサンプルを使用することで、NAS Platform を監視している場合のソリューションセットで提供している分析がアラート機能でもできます。可変値の個所は、使用する環境に応じて変更して定義してください。

```

{
  "alertDefName": "HNASAlert",
  "targets": [
    {
      "targetID": "HNASAlertTarget",
      "targetDef": [
        {
          "type": "hnas",
          "key": {
            "smuIp": "xxx.xxx.xxx.xxx"
          }
        }
      ]
    }
  ],
  "notices": [
    {
      "noticeID": "HNASAlertNotice",
      "emails": [
        {
          "addresses": [
            "test@example.com"
          ],
          "subject": "HealthCheck Alert Result",
          "body": "HNAS Alert Status is changed."
        }
      ]
    }
  ],
  "triggers": [
    {
      "triggerID": "FreePercentTrigger",
      "evaluateAllData": "N",
      "condition": {
        "warning": "%PD_HFSC.FREE_PERCENT% < '20'",
        "caution": "%PD_HFSC.FREE_PERCENT% < '45'"
      }
    },
    {
      "triggerID": "FreeCapacityPercentTrigger",
      "evaluateAllData": "N",
      "condition": {
        "warning": "%PD_HPLC.FREE_PERCENT% < '20'",
        "caution": "%PD_HPLC.FREE_PERCENT% < '30'"
      }
    }
  ],
  "definitions": [{
    "targetIDs": ["HNASAlertTarget"],
    "noticeIDs": ["HNASAlertNotice"],
    "triggerIDs": ["FreePercentTrigger", "FreeCapacityPercentTrigger"]
  }
]
}

```

Performance データベースによる仕様差異 (パフォーマンスデータを取得する場合)

パフォーマンスデータを取得する場合の Hybrid Store を使用するか Store データベースを使用するかによる Tuning Manager API の仕様差異について説明します。

- C.1 使用する Performance データベースによる仕様差異 (HTM - Agent for RAID の場合)

C.1 使用する Performance データベースによる仕様差異 (HTM - Agent for RAID の場合)

HTM - Agent for RAID では、使用する Performance データベースとして Hybrid Store を使用するか、または Store データベースを使用するかによって、Tuning Manager API の仕様には差異があります。差異がある項目と、その項目の Performance データベースごとの仕様を次の表に示します。

表 C-1 使用する Performance データベースによる仕様差異

仕様差異のある項目			Store データベースを使用する場合の仕様	Hybrid Store を使用する場合の仕様	
HTTP リクエスト	パフォーマンス データ 取得	クエリ 文字 列	startTime	レコードの RECORD_TIME フィールドに相当する時刻で検索される。	レコードの DATETIME フィールドに相当する時刻で検索される。
			endTime	レコードの RECORD_TIME フィールドに相当する時刻で検索される。	レコードの DATETIME フィールドに相当する時刻で検索される。
			フィルタ条件	ダブルクォーテーション (") で囲んだ場合に囲み文字として扱われる。 スペースを含む値をフィルタに指定する場合は、スペースを URL エンコーディングして指定することもできる。	ダブルクォーテーション (") で囲んでも無効となる (ダブルクォーテーションは単なる文字として扱われる)。 スペースを含む値をフィルタに指定する場合は、スペースを URL エンコーディングして指定する。
			granularity	指定できない。指定した場合はリクエスト不正となる。	指定できる。
			accessMode	指定できない。指定した場合はリクエスト不正となる。	指定できる。
HTTP レスポ ンス	パ フォー マンス データ 取得	レス ポンス ポ ディ	パフォー マンス データ	PI_LDS, PI_LDS1, PI_LDS2 および PI_LDS3 レコードの TOTAL_RESPONSE_RATE フィールドの要約ルールは「%」である。	PI_LDS, PI_LDS1, PI_LDS2 および PI_LDS3 レコードの TOTAL_RESPONSE_RATE フィールドの要約ルールは「HI_%」である。

このマニュアルの参考情報

このマニュアルを読むに当たっての参考情報を示します。

- [D.1 関連マニュアル](#)
- [D.2 このマニュアルでの表記](#)
- [D.3 このマニュアルで使用している略語](#)

D.1 関連マニュアル

このマニュアルの関連マニュアルを次に示します。必要に応じてお読みください。

Hitachi Tuning Manager 関連

- Hitachi Command Suite Tuning Manager 運用管理ガイド (3021-9-037)
- Hitachi Command Suite Tuning Manager インストールガイド (3021-9-038)
- Hitachi Command Suite Tuning Manager ユーザーズガイド (3021-9-039)
- Hitachi Command Suite Tuning Manager - Agents (3021-9-040)

D.2 このマニュアルでの表記

このマニュアルでは、製品名を次のように表記しています。

このマニュアルでの表記	製品名称または意味
Device Manager	Hitachi Device Manager Software
Hitachi AMS2000/AMS/WMS/SMS シリーズ	次の製品を区別する必要がない場合の表記です。 <ul style="list-style-type: none">• Hitachi Adaptable Modular Storage 2000 シリーズ• Hitachi Adaptable Modular Storage シリーズ• Hitachi Workgroup Modular Storage シリーズ• Hitachi Simple Modular Storage シリーズ
Hitachi USP	次の製品を区別する必要がない場合の表記です。 <ul style="list-style-type: none">• Hitachi Universal Storage Platform• Hitachi Network Storage Controller
Hitachi Virtual File Platform	次の製品を区別する必要がない場合の表記です。 <ul style="list-style-type: none">• Hitachi Virtual File Platform• Hitachi Capacity Optimization• Hitachi Essential NAS Platform
HTM - Agent for NAS	Hitachi Tuning Manager - Agent for NAS
HTM - Agent for RAID	Hitachi Tuning Manager - Agent for RAID
HTM - Agents	次の製品を区別する必要がない場合の表記です。 <ul style="list-style-type: none">• HTM - Agent for NAS• HTM - Agent for RAID
HUS100 シリーズ	次の製品を区別する必要がない場合の表記です。 <ul style="list-style-type: none">• Hitachi Unified Storage 150• Hitachi Unified Storage 130• Hitachi Unified Storage 110
HUS VM	Hitachi Unified Storage VM
SANRISE H シリーズ	次の製品を区別する必要がない場合の表記です。 <ul style="list-style-type: none">• Hitachi Universal Storage Platform H12000• Hitachi Universal Storage Platform H10000
Tuning Manager server	Hitachi Tuning Manager Software
Tuning Manager シリーズ	Tuning Manager server および HTM - Agents の総称です。
Universal Storage Platform V/VM シリーズ	次の製品を区別する必要がない場合の表記です。 <ul style="list-style-type: none">• Hitachi Universal Storage Platform V• Hitachi Universal Storage Platform VM• Hitachi Universal Storage Platform H24000• Hitachi Universal Storage Platform H20000
Virtual Storage Platform シリーズ	次の製品を区別する必要がない場合の表記です。

このマニュアルでの表記	製品名称または意味
	<ul style="list-style-type: none"> Hitachi Virtual Storage Platform Hitachi Virtual Storage Platform VP9500
VSP 5000 シリーズ	次の製品を区別する必要がない場合の表記です。 <ul style="list-style-type: none"> Hitachi Virtual Storage Platform 5100 Hitachi Virtual Storage Platform 5500 Hitachi Virtual Storage Platform 5100H Hitachi Virtual Storage Platform 5500H
VSP E990	Hitachi Virtual Storage Platform E990
VSP Fx00 モデル	次の製品を区別する必要がない場合の表記です。 <ul style="list-style-type: none"> Hitachi Virtual Storage Platform F350 Hitachi Virtual Storage Platform F370 Hitachi Virtual Storage Platform F400 Hitachi Virtual Storage Platform F600 Hitachi Virtual Storage Platform F700 Hitachi Virtual Storage Platform F800 Hitachi Virtual Storage Platform F900
VSP F1500	Hitachi Virtual Storage Platform F1500
VSP Gx00 モデル	次の製品を区別する必要がない場合の表記です。 <ul style="list-style-type: none"> Hitachi Virtual Storage Platform G100 Hitachi Virtual Storage Platform G130 Hitachi Virtual Storage Platform G150 Hitachi Virtual Storage Platform G200 Hitachi Virtual Storage Platform G350 Hitachi Virtual Storage Platform G370 Hitachi Virtual Storage Platform G400 Hitachi Virtual Storage Platform G600 Hitachi Virtual Storage Platform G700 Hitachi Virtual Storage Platform G800 Hitachi Virtual Storage Platform G900
VSP G1000	次の製品を区別する必要がない場合の表記です。 <ul style="list-style-type: none"> Hitachi Virtual Storage Platform G1000 Hitachi Virtual Storage Platform VX7
VSP G1500	Hitachi Virtual Storage Platform G1500

D.3 このマニュアルで使用している略語

このマニュアルで使用している略語を次の表に示します。

略語	正式名称
ABNF	Augmented Backus-Naur form
API	Application Programming Interface
CSV	Comma Separated Value
EVS	Enterprise Virtual Server
GUI	Graphical User Interface
ID	IDentifier
IP	Internet Protocol
JSON	JavaScript Object Notation
KDC	Key Distribution Center
LDAP	Lightweight Directory Access Protocol

略語	正式名称
Linux	Tuning Manager server がサポートしている Red Hat Enterprise Linux(R)および Oracle Linux(R)の総称です。
OS	Operating System
RADIUS	Remote Authentication Dial In User Service
REST	Representational State Transfer
SMU	System Management Unit
SSL	Secure Sockets Layer
SSO	Single Sign-On
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UTC	Coordinated Universal Time
XML	eXtensible Markup Language

索引

A

Action オブジェクト 33
API のバージョン 31

B

Basic 認証 30

C

Collection オブジェクト 32

E

Error オブジェクト 32

H

HTM - Agent for RAID のインスタンス一覧の取得 85
HTM - Agents 一覧の手動リフレッシュ処理実行時のパラメーターの取得 75
HTM - Agents 一覧の手動リフレッシュ処理の実行 76
HTM - Agents 管理のアクション一覧の取得 99
HTM - Agents のインスタンス一覧の取得 79
HTTP メソッド 28

L

LDEV 性能の Top5 のチャートを作成する手順 154

P

PrimitiveValue オブジェクト 33

T

Tuning Manager API の共通仕様 26
Tuning Manager API の仕様 25

U

URI の構成要素 26

あ

アクション一覧の取得 99
圧縮転送 31

い

インスタンス一覧の取得 85
インストール (API の場合) 21

き

共通オブジェクト 32

こ

更新できる情報 20
構成する項目 (リクエストライン) 28
構成要素 (URI) 26

さ

サポートしている HTTP メソッド 28

し

取得できる情報 19

せ

- セキュリティ 30
- セットアップ (API の場合) 21

に

- 認証 30

は

- バージョン (API) 31
- パフォーマンスデータの取得 87

へ

- ヘッダー 29

め

- メタ情報取得 104
- メッセージ一覧 115
- メッセージの形式 114
- メッセージの出力先一覧 114

り

- リクエストメッセージ 28
- リクエストライン 28

 株式会社 日立製作所

〒 100-8280 東京都千代田区丸の内一丁目 6 番 6 号
