



HA8000 シリーズ

BladeSymphony

Hitachi Server Navigator ユーザーズガイド

FAST FIND LINKS

[ドキュメント構成](#)

[お問い合わせ先](#)

[目次](#)

Hitachi, Ltd.

ソフトウェア使用上の注意

お客様各位

株式会社 日立製作所

このたびは BladeSymphony または日立アドバンストサーバ HA8000 をお買い上げいただき、誠にありがとうございます。

下記の「ソフトウェアの使用条件」を必ずお読みいただきご了解いただきますようお願いいたします。ソフトウェアの使用条件

1. ソフトウェアの使用

このソフトウェアは、特定の 1 台の BladeSymphony サーバブレードまたは日立アドバンストサーバシステムでのみ使用することができます。

2. 複製

お客様は、このソフトウェアの一部または全部の複製を行わないでください。ただし、下記に該当する場合に限り複製することができます。

お客様がご自身のバックアップ用、保守用として、1 項に定める 1 台の BladeSymphony サーバブレードまたは日立アドバンストサーバシステムで使用する場合に限り複製することができます。

3. 改造・変更

お客様によるこのソフトウェアの改造・変更は行わないでください。万一、お客様によりこのソフトウェアの改造・変更が行われた場合、弊社は該当ソフトウェアについてのいかなる責任も負いません。

4. 第三者の使用

このソフトウェアを譲渡、貸出、移転その他の方法で、第三者に使用させないでください。

5. 保証の範囲

- (1) 万一、媒体不良のために、ご購入時に正常に機能しない場合には、無償で交換いたします。
- (2) このソフトウェアの使用により、万一お客様に損害が生じたとしても、弊社は責任を負いません。あらかじめご了承ください。

以上

安全にお使いいただくために

安全に関する注意事項は、下に示す見出しによって表示されます。これは安全警告記号と「警告」、「注意」および「通知」という見出し語を組み合わせたものです。



⚠ 警告

⚠ 注意

これは、安全警告記号です。人への危害を引き起こす潜在的な危険に注意を喚起するために用います。起こりうる傷害または死を回避するために、このシンボルのあとに続く安全に関するメッセージにしたがってください。

これは、死亡または重大な傷害を引き起こすおそれのある潜在的な危険の存在を示すのに用います。

これは、軽度の傷害、あるいは中程度の傷害を引き起こすおそれのある潜在的な危険の存在を示すのに用います。

NOTICE 通知

これは、人身障害とは関係のない損害を引き起こすおそれのある場合に用います。



【表記例 1】感電注意

△の図記号は注意していただきたいことを示し、△の中に「感電注意」などの注意事項の絵が描かれています。



【表記例 2】分解禁止

○の図記号は行ってはいけないことを示し、○の中に「分解禁止」などの禁止事項の絵が描かれています。

なお、○の中に絵がないものは、一般的な禁止事項を示します。



【表記例 3】電源プラグをコンセントから抜け

●の図記号は行っていただきたいことを示し、●の中に「電源プラグをコンセントから抜け」などの強制事項の絵が描かれています。

なお、●は一般的に行っていただきたい事項を示します。

安全に関する共通的な注意について

次に述べられている安全上の説明をよく読み、十分理解してください。

- 操作は、このマニュアル内の指示、手順に従って行ってください。
- 本製品やマニュアルに表示されている注意事項は必ず守ってください。
- 本ソフトウェアをインストールするシステム装置のマニュアルを参照し、記載されている注意事項は必ず守ってください。
これを怠ると、人身上の傷害やシステムを含む財産の破損を引き起こすことがあります。

操作や動作は

マニュアルに記載されている以外の操作や動作は行わないでください。

本製品について何か問題がある場合は、お買い求め先にご連絡いただくか保守員をお呼びください。

自分自身でもご注意を

本製品やマニュアルに表示されている注意事項は、十分検討されたものです。それでも、予測を超えた事態が起こることが考えられます。操作に当たっては、指示に従うだけでなく、常に自分自身でも注意するようにしてください。

安全にお使いいただくために（続き）

製品の損害を防ぐための注意

本製品の取り扱いにあたり次の注意事項を常に守ってください。



本製品のインストールについて

本製品は、本製品の動作をサポートしているシステム装置でご使用ください。それ以外のシステム装置にインストールすると、システム装置の仕様の違いにより故障の原因となります。サポートの有無については、システム装置のマニュアルなどでご確認ください。

本マニュアル内の警告表示

⚠️ 警告

本マニュアル内にはありません。

⚠️ 注意

本マニュアル内にはありません。

NOTICE 通知

本マニュアル内にはありません。

目次

安全にお使いいただくために	iii
目次	v
お使いになる前に	vii
重要なお知らせ	viii
規制・対策などについて	viii
登録商標・商標について	viii
版権について	ix
文書来歴	ix
ドキュメント構成	ix
オペレーティングシステム(OS)の略称について	x
凡例	xii
お問い合わせ先	xiii
Server Navigatorの概要	1-1
概要	1-2
機能と動作に必要なシステム環境	1-6
制限事項	1-13
Server Navigatorのインストール	2-1
インストール	2-2
アップグレード	2-7
アンインストール	2-9

Server NavigatorのGUI画面の説明	3-1
Server Navigator GUIの起動	3-2
アップデート機能用GUI画面	3-5
監視機能設定機能用GUI画面	3-23
ツール連携機能用GUI画面	3-25
Server NavigatorのGUI操作の説明	4-1
アップデート機能のGUI操作	4-2
手動ダウンロード実行	4-8
自動アップデート設定	4-13
手動アップデート実行	4-17
手動ロールバック実行	4-22
監視機能設定機能のGUI操作	4-28
ツール連携機能のGUI操作	4-29
Server NavigatorのCLI操作の説明	5-1
Server NavigatorのCLI操作概要	5-2
Server NavigatorのCLI操作例一覧	5-2
トラブルシューティング	6-1
トラブルシートの概要	6-2
インストールメッセージとファイル一覧	A-1
インストール/アンインストール時のメッセージ一覧	A-2
インストールファイル一覧	A-4
CLI詳細	B-1
Server Navigator CLIのメソッド一覧	B-2
Server Navigator CLIで使用するクエリー一覧	B-31
Server NavigatorのCLIサンプル一覧	B-35
頭字語と略語	略語-1

使いになる前に

このマニュアルは、弊社サーバ製品を使用する前に、知っておいていただきたい内容について説明しています。製品を使用する前に、安全上の指示をよく読み十分理解してください。このマニュアルは、いつでも参照できるよう、手近な所に保管してください。

この章の内容は以下の通りとなっています:

- [重要なお知らせ](#)
- [規制・対策などについて](#)
- [登録商標・商標について](#)
- [版権について](#)
- [文書来歴](#)
- [ドキュメント構成](#)
- [オペレーティングシステム\(OS\)の略称について](#)
- [凡例](#)
- [お問い合わせ先](#)



弊社サーバ製品の使用は、弊社とのお客様の契約の条件によって決定されます。

重要なお知らせ

- 本書の内容の一部、または全部を無断で転載したり、複写することは固くお断りします。
- 本書の内容について、改良のため予告なしに変更することがあります。
- 本書の内容については万全を期しておりますが、万一ご不審な点や誤りなど、お気付きのことがありましたら、お買い求め先へご一報くださいますようお願ひいたします。
- 本書に準じないで本製品を運用した結果については責任を負いません。あらかじめご了承ください。

規制・対策などについて

輸出規制について

本製品を輸出される場合には、外国為替および外国貿易法並びに米国の輸出管理関連法規などの規制をご確認の上、必要な手続きをお取りください。なお、ご不明の場合は弊社担当営業にお問い合わせください。

海外での使用について

本装置は日本国内専用です。国外では使用しないでください。なお、他国には各々の国で必要となる法律、規格等が定められており、本装置は適合していません。

登録商標・商標について

Microsoft、Windows、Windows Server、Hyper-V 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。

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。

Red Hat は、米国およびその他の国における Red Hat Inc. の商標または登録商標です。

LSI および LSI Logic のロゴは LSI Corporation の商標で何らかの司法権に登録されている場合があります。

インテル、Intel、Xeon はアメリカ合衆国およびその他の国における Intel Corporation の商標です。

その他、本マニュアル中の製品名および会社名は、各社の商標または登録商標です。

版権について

このマニュアルの内容はすべて著作権によって保護されています。このマニュアルの内容の一部または全部を無断で記載することは禁じられています。

Copyright © Hitachi, Ltd. 2012, 2013. All rights reserved.

文書来歴

改訂	年月日	説明
SNV-2_1	2012年11月	初版
SNV-2_2	2012年12月	第2版
SNV-2_3	2012年12月	第3版
SNV-2_4	2012年12月	第4版
SNV-2_5	2013年1月	第5版

ドキュメント構成

このドキュメントの内容と構成の概要を下記表に示します。各章のタイトルをクリックすることで、各章を参照することができます。

章/付録	説明
Chapter 1, Server Navigator の概要	この章では Server Navigator の概要について説明します。
Chapter 2, Server Navigator のインストール	この章では、Server Navigator のインストール方法およびアンインストール方法について説明します。
Chapter 3, Server Navigator のGUI画面の説明	この章では、Server Navigator の GUI 画面について説明します。
Chapter 4, Server Navigator のGUI操作の説明	この章では、Server Navigator GUI を使用した各種ユースケースにおける作業手順について説明します。
Chapter 5, Server Navigator のCLI操作の説明	この章では、Server Navigator の GUI 操作を CLI での実行に置き換えた例について説明します。
Chapter 6, トラブルシューティング	この章では、Server Navigator のトラブルシューティングについて説明します。
Appendix A, インストールメッセージとファイル一覧	この付録では、メッセージ一覧とインストールファイル一覧について説明します。
Appendix B, CLI詳細	この付録では、Server Navigator の CLI の詳細について説明します。

オペレーティングシステム(OS)の略称について

本マニュアルでは、次の OS 名称を省略して表記します。

- Microsoft® Windows Server® 2012, Datacenter Edition 日本語版
(以下 Windows Server 2012, Datacenter Edition または
Windows Server 2012)
- Microsoft® Windows Server® 2012, Standard Edition 日本語版
(以下 Windows Server 2012, Standard Edition または
Windows Server 2012)
- Microsoft® Windows Server® 2008 R2, Standard x64 Edition 日本語版
(以下 Windows Server 2008 R2, Standard x64 Edition または
Windows Server 2008 R2 x64 Editions、Windows Server 2008 R2)
- Microsoft® Windows Server® 2008 R2, Enterprise x64 Edition 日本語版
(以下 Windows Server 2008 R2, Enterprise x64 Edition または
Windows Server 2008 R2 x64 Editions、Windows Server 2008 R2)
- Microsoft® Windows Server® 2008 Standard 日本語版
(以下 Windows Server 2008 Standard または Windows Server 2008、Windows)
- Microsoft® Windows Server® 2008 Enterprise 日本語版
(以下 Windows Server 2008 Enterprise または Windows Server 2008、Windows)
- Microsoft® Windows Server® 2008 Standard without Hyper-V™ 日本語版
(以下 Windows Server 2008 Standard without Hyper-V または
Windows Server 2008 Standard、Windows Server 2008、Windows)
- Microsoft® Windows Server® 2008 Enterprise without Hyper-V™ 日本語版
(以下 Windows Server 2008 Enterprise without Hyper-V または
Windows Server 2008 Enterprise、Windows Server 2008、Windows)
- Microsoft® Windows Server® 2003 R2, Standard x64 Edition 日本語版
(以下 Windows Server 2003 R2, Standard x64 Edition または
Windows Server 2003 R2 x64 Editions、Windows Server 2003 R2)
- Microsoft® Windows Server® 2003 R2, Enterprise x64 Edition 日本語版
(以下 Windows Server 2003 R2, Enterprise x64 Edition または
Windows Server 2003 R2 x64 Editions、Windows Server 2003 R2)
- Microsoft® Windows Server® 2003 R2, Standard Edition 日本語版
(以下 Windows Server 2003 R2, Standard Edition または
Windows Server 2003 R2(32 ビット)、Windows Server 2003、R2)
- Microsoft® Windows Server® 2003 R2, Enterprise Edition 日本語版
(以下 Windows Server 2003 R2, Enterprise Edition または
Windows Server 2003 R2(32 ビット)、Windows Server 2003、R2)

- Microsoft®Windows Server® 2003, Standard x64 Edition 日本語版
(以下 Windows Server 2003, Standard x64 Edition または
Windows Server 2003 x64 Editions、Windows Server 2003、Windows)
- Microsoft®Windows Server® 2003, Enterprise x64 Edition 日本語版
(以下 Windows Server 2003, Enterprise x64 Edition または
Windows Server 2003 x64 Editions、Windows Server 2003、Windows)
- Microsoft®Windows Server® 2003, Standard Edition 日本語版
(以下 Windows Server 2003, Standard Edition または
Windows Server 2003(32 ビット)、Windows Server 2003、Windows)
- Microsoft®Windows Server® 2003, Enterprise Edition 日本語版
(以下 Windows Server 2003, Enterprise Edition または
Windows Server 2003(32 ビット)、Windows Server 2003、Windows)

凡例

弊社サーバ製品の用語は特に明記がない場合、弊社サーバ製品すべてのモデルで使用されています。このドキュメントで使用されている記号は以下の通りです。

記号	説明
太字	メニュー、オプション、ボタン、フィールドおよびラベルを含めて、ウィンドウ・タイトル以外に表示される内容を示します。 例: Click OK .
イタリック体	ユーザまたはシステムによって提供される変数を示します。 例: copy source-file target-file 通知: "<>" も変数を示すために使用されます。
画面/コマンドライン	画面に表示またはユーザによって入力する内容を示します。 例: # pairdisplay -g oradb
< >	ユーザまたはシステムによって提供される変数を示します。 例: # pairdisplay -g <group> 通知: イタリック体のフォントも変数を示すために使用されます。
[]	オプションの値を示します。 例: [a b] a, b または入力なしのどちらかを選択することを示します。
{ }	必要な値あるいは予期された値を示します。 例: { a b } a または b のどちらかを選択することを示します。
 	2つ以上のオプションあるいは引数から選択できることを示します。 例: [a b] a, b または入力なしのどちらかを選択することを示します。 { a b } a または b のどちらかを選択することを示します。
アンダーライン	デフォルト値を示します。例: <u>a</u> b

このドキュメントは、注意すべき情報に対して次のアイコンを使用しています:

アイコン	意味	記述
	警告	死亡または重大な傷害を引き起こすおそれのある潜在的な危険の存在を示すのに用います。
	注意	軽度の傷害、あるいは中程度の傷害を引き起こすおそれのある潜在的な危険の存在を示すのに用います。
	通知	人身傷害とは関係のない損害を引き起こすおそれのある場合に用います。
	制限	本製品の故障や障害の発生を防止し、正常に動作させるための事項を示します。
	補足	本製品を活用するためのアドバイスを示します。

お問い合わせ先

HA8000 にて使用時のお問い合わせ先

技術情報、アップデートプログラムについて

HA8000 ホームページで、技術情報、ドライバやユーティリティ、BIOS/EFI、ファームウェアなどのアップデートプログラムを提供しております。本ユーティリティでアップデートに対応していない場合やトラブルシューティングで必要となります。[ダウンロード]をクリックしてください。

- HA8000 ホームページアドレス : <http://www.hitachi.co.jp/ha8000>

各アップデートプログラムの適用はお客様責任にて実施していただきますが、システム装置を安全にご使用いただくためにも、定期的にホームページにアクセスして、本ユーティリティを使用して最新のドライバやユーティリティ、BIOS/EFI、ファームウェアへ更新していただくことをお勧めいたします。

障害等の保守作業で部品を交換した場合、交換した部品の BIOS/EFI、ファームウェアは原則として最新のものが適用されます。また保守作業時、交換していない部品の BIOS/EFI、ファームウェアも最新のものへ更新する場合があります。

なお、お客様による BIOS/EFI、ファームウェアアップデート作業が困難な場合は、有償でアップデート作業を代行するサービスを提供いたします。詳細はお買い求め先にお問い合わせください。

操作や使いこなしについて

本製品のハードウェアについての機能や操作方法に関するお問い合わせは、HCA センタ (HITAC カスタマ・アンサ・センタ) でご回答いたしますので、次のフリーダイヤルにおかけください。受付担当がお問い合わせ内容を承り、専門エンジニアが折り返し電話でお答えするコールバック方式をとらせていただきます。

HCA センタ (HITAC カスタマ・アンサ・センタ)

 0120-2580-91

受付時間

9:00～12:00/13:00～17:00 (土・日・祝日、年末年始を除く)

お願い

- お問い合わせになる際に次の内容をメモし、お伝えください。お問い合わせ内容の確認をスムーズに行うため、ご協力をお願いいたします。

形名 (TYPE) ／ 製造番号 (S/N) ／ インストール OS ／ サービス ID (SID)

「形名」、「製造番号」および「サービス ID」は、システム装置前面に貼り付けられている機器ラベルにてご確認ください。

- 質問内容を FAX でお送りいただくこともありますので、ご協力をお願ひいたします。
- HITAC カスタマ・アンサ・センタでお答えできるのは、製品のハードウェアの機能や操作方法などです。ハードウェアに関する技術支援や、OS や各言語によるユーザープログラムの技術支援は除きます。
ハードウェアやOSの技術的なお問い合わせについては有償サポートサービスにて承ります。詳細は、「[技術支援サービスについて](#)」(P.xiv)をご参照ください。
- 明らかにハードウェア障害と思われる場合は、販売会社または保守会社にご連絡ください。

欠品・初期不良・故障について

本製品の納入時の欠品や初期不良および修理に関するお問い合わせは日立コールセンタにご連絡ください。

日立コールセンタ

 0120-921-789

受付時間

9:00～18:00 (土・日・祝日、年末年始を除く)

お願い

- お電話の際には、製品同梱の保証書をご用意ください。
- Webによるお問い合わせは次へお願いします。
https://e-biz.hitachi.co.jp/cgi-shell/qa/rep_form.pl?TXT_MACTYPE=1

技術支援サービスについて

ハードウェアやソフトウェアの技術的なお問い合わせについては、技術支援サービスによる有償サポートとなります。

日立統合サポートサービス「日立サポート 360」

ハードウェアと、Windows や Linux などの OS を一体化したサポートサービスをご提供いたします。

詳細は次の URL で紹介しています。

- ホームページアドレス

<http://www.hitachi.co.jp/soft/symphony/>

インストールや運用時のお問い合わせや問題解決など、システムの円滑な運用のためにサービスのご契約をお勧めします。

HA8000 問題切分支援・情報提供サービス

ハードウェアとソフトウェアの問題切り分け支援により、システム管理者の負担を軽減します。 詳細は次の URL で紹介しています。

- ホームページアドレス

<http://www.hitachi.co.jp/soft/HA8000/>

運用時の問題解決をスムーズに行うためにサービスのご契約をお勧めします。

なお、本サービスには OS の技術支援サービスは含まれません。OS の技術支援サービスを必要とされる場合は「日立サポート 360」のご契約をお勧めします。

BladeSymphonyにて使用時のお問い合わせ先

最新情報・Q&A・ダウンロードは

「BladeSymphony ホームページ」で、重要なお知らせ、Q&A やダウンロードなどの最新情報を提供しております。各アップデートプログラムの適用はお客様責任にて実施していただきますが、システム装置を安全にご使用いただくためにも、定期的にホームページにアクセスして、最新のドライバやユーティリティ、BIOS、ファームウェアへ更新していただくことをお勧めいたします。

- ホームページアドレス : <http://www.hitachi.co.jp/products/bladesymphony/>

- ・重要なお知らせ

BladeSymphony の使用における重要なお知らせを掲載しています。

- ・Q & A (よくあるご質問)

BladeSymphony に関するよくあるご質問とその回答を掲載しています。

[製品] タブをクリックし、画面右の [Q&A よくあるご質問] をクリックしてください。

- ・ドライバ・ユーティリティ ダウンロード

修正モジュール/ ドライバ/ ファームウェア/ ユーティリティなどの最新情報を提供しています。 [サポート&ダウンロード] タブをクリックし、「ドライバ・ユーティリティ ダウンロード」の [詳細はこちる] をクリックしてください。

- ・マニュアル

製品添付マニュアル(ユーザーズガイド)の最新情報を提供しています。

[サポート & ダウンロード] タブをクリックし、「マニュアル」の [詳細はこちら] をクリックしてください。

困ったときは

1. マニュアルを参照してください。製品同梱の他の紙マニュアルもご利用ください。
2. 電話でお問い合わせください。
 - ・販売会社からご購入いただいた場合
販売会社で修理を承ることがございます。お買い求め先へ修理の窓口をご確認ください。
 - ・上記以外の場合
日立ソリューションサポートセンタまでお問い合わせください。

日立ソリューションサポートセンタ

- ・BladeSymphony サポートセンタ
フリーダイヤル：サポートサービス契約の締結後、別途ご連絡いたします。
詳細は担当営業までお問い合わせください。
- 受付時間 : 8:00～19:00
(土・日・祝日・年末年始を除く)

Server Navigatorの概要

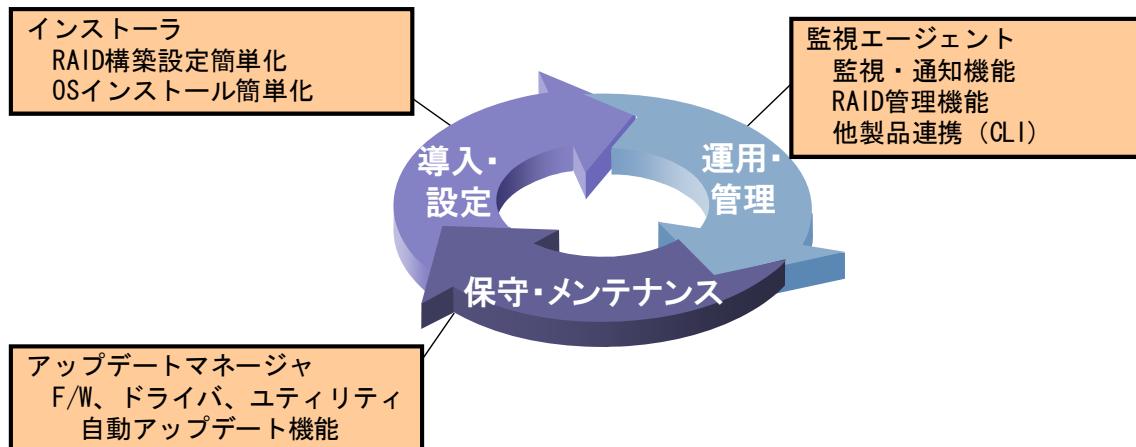
この章では Server Navigator の概要について説明します。

- [概要](#)
- [機能と動作に必要なシステム環境](#)
- [制限事項](#)

概要

サーバに標準添付しているツールキット「Hitachi Server Navigator」は、OS/ドライバの自動インストールや、ユーティリティ/ドライバ/ファームウェアの自動アップデート機能を提供します。本機能によって、従来と比較してOSインストール時およびドライバアップデート時の作業手順を簡略化することができ、お客様の作業負担を軽減します。

Server Navigator の概念図は以下になります。



インストール機能(インストーラ)

管理対象サーバの導入を支援します。

事前のパラメーター括入力によるOSやドライバを自動インストールすることができます。

また、内蔵ディスクのRAID構築がシンプルで分かりやすいGUIにより、従来に比べて、少ない操作数で簡単に行うことができます。

監視機能(監視エージェント)

管理対象サーバの監視を支援します。

OS上のログ情報などを監視することで、ハードウェアの障害を検知・通知し、迅速な障害復旧が可能になります。また、サーバ内蔵ディスクアレイの状態監視、障害通知も行うことができ、装置の稼働率向上を実現します。

アップデート機能(アップデートマネージャ)

管理対象サーバのメンテナンスを支援します。

GUIによる簡単な画面操作で、ファームウェア、ドライバ、ユーティリティアップデートを行うことが可能です。自動ダウンロード機能、および、自動アップデート機能を使用することにより、スケジュール設定後は操作不要な運用が可能です。



本ドキュメントでは、各用語を次のように定義します。

- ダウンロード：
Server Navigator で保管するリポジトリへアップデートモジュールを格納すること
- ロールバック：
適用したアップデートを元に戻すアップデートを行うこと
- ダウングレード：
バージョンダウンとなるアップデートまたはロールバックを行うこと
- コンポーネント：
管理対象である、F/W、ドライバ、ユーティリティ
- リポジトリ：
ダウンロードしたアップデートモジュールを保管する場所
- CIM：
CIM は管理対象となるデバイスやソフトウェアを抽象化して表現するモデルで、Windows では WMI
- CIM Provider/プロバイダ：
CIM で抽象化したデバイスやソフトウェアへのアクセスを提供するソフトウェア
- CLI：
文字による入出力を行うユーザインタフェース（コマンドラインインタフェース）
- クラス：
オブジェクト指向においてオブジェクトの設計図にあたり、データ（メンバ変数）と操作（メソッド）が記述される
- メソッド：
あるクラスに所属するサブルーチン
- クエリ：
WMI クエリ言語を示し、指定のオブジェクトや操作するオブジェクトを取得するのに使用

関連ドキュメント

HA8000 シリーズ/BladeSymphony
Hitachi Server Navigator ユーザーズガイド [本書]

HA8000 シリーズ/BladeSymphony
Hitachi Server Navigator OS セットアップガイド

HA8000 シリーズ/BladeSymphony
Hitachi Server Navigator ユーザーズガイド アップデート機能

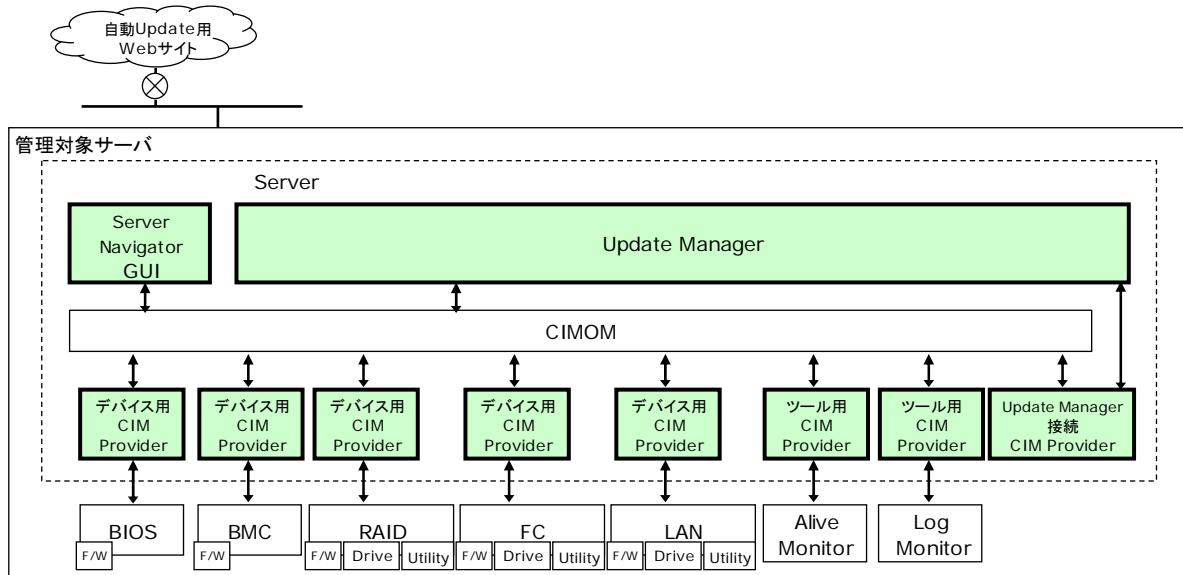
HA8000 シリーズ/BladeSymphony
Hitachi Server Navigator ユーザーズガイド Log Monitor 機能

HA8000 シリーズ/BladeSymphony
Hitachi Server Navigator ユーザーズガイド Alive Monitor 機能

HA8000 シリーズ/BladeSymphony
Hitachi Server Navigator ユーザーズガイド RAID 管理機能

構成

Server Navigator の構成は以下になります。



以下に、構成図の各部の概要を説明します。

部位	概要
Update Manager	ダウンロードおよびアップデートの管理を行います。OS上でサービスとして提供され、ログオンしない状態でもバックグラウンドで動作します。
Update Manager 接続 CIM Provider	管理対象デバイスおよび Update Manager を CIM 基盤により抽象化します。インターフェースとして機能します。
デバイス用 CIM Provider	要求に従い接続デバイスの状態管理およびアップデートを行います。
ツール用 CIM Provider	要求に従い接続連携ツールの状態管理およびアップデートを行います。
CIMOM	CIM Provider と管理アプリケーション間のコミュニケーションを管理します。
Server Navigator GUI	管理対象サーバ上で動作する、本ユーティリティの GUI です。操作には管理者権限が必要です。
BIOS/EFI、BMC、RAID、FC、LAN	管理対象デバイスです。
F/W、Driver、Utility	管理対象デバイスの部位です。
Alive Monitor、Log Monitor	管理対象ツールです。

機能と動作に必要なシステム環境

Server Navigatorの機能

Server Navigator では、管理対象コンポーネントに対して次の機能を GUI/CLI 操作で提供します。以降、対象コンポーネントとは、管理対象サーバ上の管理対象デバイスの部位である F/W、Driver、Utility と、管理対象ツールと、Server Navigator 自身を指します。

アップデート機能

管理対象サーバのファームウェア、ドライバ、ユティリティを管理し、アップデートする機能です。

アップデート機能の詳細は、「Hitachi Server Navigator ユーザーズガイド アップデート機能」を参照してください。

監視機能

管理対象サーバのハードウェアおよび OS を監視する機能です。

ハードウェア監視機能の詳細は、「Hitachi Server Navigator ユーザーズガイド Log Monitor 機能」を参照してください。

OS 監視機能の詳細は、「Hitachi Server Navigator ユーザーズガイド Alive Monitor 機能」を参照してください。

Server Navigatorによる設定方法は、「[監視機能設定機能用GUI画面](#)」を参照してください。

RAID 管理機能

管理対象サーバのハードおよびソフト RAID を管理する機能です。Server Navigator のツール連携機能から起動することができます。

RAID 管理機能の詳細は、「Hitachi Server Navigator ユーザーズガイド RAID 管理機能」を参照してください。

ツール連携機能

Server Navigator と連携するツールを呼び出す機能です。

ツール連携機能の詳細は、「[ツール連携機能用GUI画面](#)」を参照してください。

トレースログ機能

アップデート機能およびツール連携機能のトレースログを採取する機能です。

Server Navigator GUI

直感的なユーザインターフェースで、Server Navigator が備える機能を使用することができます。ダウンロードまたはアップデートまたはロールバック中は進行状況の確認が行えます。

トレースログを閲覧することにより、各機能の実行結果の確認および問題発生時のトラブルシューティングが可能です。

Server Navigator CLI

WMI から Server Navigator の各種 CIM Provider で定義されるクラスのメソッドを呼び出すことにより、Server Navigator が備える各種機能を CLI で実行することができます。

管理インターフェースを公開していますので、他製品との連携が容易になります。

各種処理時間の目安



「[各種処理時間の目安](#)」(P.1-8)や「[要求数量の目安](#)」(P.1-10)に記載している評価データは下記構成で測定しております。

サーバ装置 : HA-8000/RS-210

CPU : Intel(R) Xeon(R) CPU E5-2430 2.2GHz

メモリ: 4GB

OS : Windows Server 2008 Enterprise

システム装置が無負荷の状態における Server Navigator のダウンロードおよびアップデートの処理時間の目安を以下に表記します。

機能	対象コンポーネント	処理時間 *
自動ダウンロードまたは一括ダウンロード	共通	ダウンロードするモジュールの種類 × 1 分が目安ですが、モジュールの大きさや回線状況によっては数十分かかる場合もあります。
アップデートまたは一括アップデートまたは一括ロールバック	BIOS/EFI	約 70 秒 + リブート時間
	BMC	約 750 秒 + リブート時間 *
	LSI MegaRAID SAS F/W	約 180 秒 + リブート時間
	LSI MegaRAID SAS Driver	約 40 秒 + リブート時間
	RAID Utility(MSM)	約 120 秒 + リブート時間
	RAID Utility(HRN)	約 40 秒 + リブート時間
	FC (Hitachi) F/W	約 90 秒 + リブート時間
	FC (Hitachi) Driver	約 40 秒 + リブート時間
	FC (Hitachi) Utility	約 40 秒 + リブート時間
	LAN (Broadcom) F/W	約 100 秒 + リブート時間
	LAN (Broadcom) Driver	約 60 秒 + リブート時間
	LAN (Broadcom) Utility	約 40 秒 + リブート時間
	Log Monitor	約 40 秒 + リブート時間
	Alive Monitor	約 40 秒 + リブート時間
	Server Navigator	約 40 秒 + リブート時間

* システムリブート中、BMC FW のアップデート反映処理による BMC 再起動が実行されるため、通常より約 140 秒リブート時間が長くなります。

システムリブート回数の目安

自動アップデートまたは手動アップデートまたは手動ロールバック実行時のシステムリブート回数の目安を以下に表記します。

対象コンポーネント	リブート回数 *1			
	アップデート前処理工程	アップデート工程	リストア工程	合計
BIOS/EFI	0回	1回	0回	1回
BMC	0回	1回	0回	1回
LSI MegaRAID SAS F/W	0回	1回	0回	1回
LSI MegaRAID SAS Driver	0回	1回	0回	1回
Soft RAID Driver	0回	1回	0回	1回
RAID Utility(MSM)	1回	1回	0回	2回
RAID Utility(HRN)	0/1回 *2	1回	0回	1/2回 *2
FC (Hitachi) F/W	1回	1回	0回	2回
FC (Hitachi) Driver	1回	1回	0回	2回
FC (Hitachi) Utility	1回	1回	0回	2回
LAN (Broadcom) F/W	1回	1回	1回	3回
LAN (Broadcom) Driver	1回	1回	1回	3回
LAN (Broadcom) Utility	1回	1回	1回	3回
Log Monitor	0回	1回	0回	1回
Alive Monitor	0回	1回	0回	1回
Server Navigator	0回	1回	0回	1回

*1 アップデート処理およびロールバック処理では、アップデート前処理工程、アップデート工程、リストア工程の完了時に必要に応じシステムリブートを行います。複数コンポーネントの場合でも、同時に各工程を進めるため、システムリブート回数は最大3回です。ただし、Server Navigatorのアップデートはデバイス/ツールのアップデートに先行して行うため、この場合最大4回となります。

*2 ダウングレード時のみシステムリブートを行います。

動作に必要なシステム環境

Server Navigator は、以下の条件を満たす環境で動作します。

項目	仕様
サポート機種	BS500 BS2000 x3 モデル *1 BS320 x6 モデル *1 *3 HA8000 xM/xM1/xL2 モデル *1
OS *2	Windows Server 2012 Standard Windows Server 2012 Datacenter Windows Server 2008 R2 Standard Windows Server 2008 R2 Enterprise Windows Server 2008 R2 Datacenter Windows Server 2008 Standard Windows Server 2008 Enterprise Windows Server 2008 Datacenter Windows Server 2008 Standard without Hyper-V Windows Server 2008 Enterprise without Hyper-V Windows Server 2008 Datacenter without Hyper-V Red hat Enterprise Linux 6 Red hat Enterprise Linux 5
CPU	動作クロック: 2 GHz 以上推奨
メモリ	512 MB 以上: 2 GB 以上推奨
ネットワークインターフェース	100Mbps 以上 (Web サイトからダウンロードを行う場合)
HDD 等	2GB 以上の空き容量
画面出力	解像度: 1024 x 768 以上

*1 BS2000、BS320、RS440 xL2 モデルは、アップデート機能のサポート対象外です。
*2 Server Navigator を CLI によりリモート管理を行う場合、CLI 実行元である管理サーバの OS も同条件です。
Red hat Enterprise Linux は、アップデート機能のサポート対象外です。
*3 BS320 は、Alive Monitor 機能のサポート対象外です。

要求リソース

Server Navigator は常駐型アプリケーションです。以下に、無負荷時、ダウンロード時、アップデート時における必要なシステム要求リソースの目安を表記します。

項目	無負荷時	ダウンロード時 *1	アップデート時 *2
CPU 負荷率	0%	18%	13%
メモリ使用量	約 40MB	約 100MB	約 170MB

項目	無負荷時	ダウンロード時 *1	アップデート時 *2
*1 ダウンロード時とは、自動ダウンロード機能、ダウンロード指示機能によりダウンロード処理を行っている場合を示します。			
*2 アップデート時とは、自動アップデート機能、アップデート指示機能、ロールバック指示機能によりアップデート処理を行っている場合を示します。CPU 負荷率は最大値で、その場合システムが一時的に遅くなる場合がありますが仕様です。			
CPU 負荷率の表示に限らず一時的に応答が遅くなる場合がありますが問題ございません。			

Server Navigator は以下のサービスが常駐します。

サービス名称	説明
UpdateManagerService	Server Navigator のダウンロード機能およびアップデート機能を提供します。 UpdateManagerService と依存性のあるサービスは WMI サービス(Winmgmt)です。



UpdateManagerService は WMI サービス (Winmgnt) と依存性があります。WMI サービス開始前に UpdateManagerService を開始しないでください。

Server Navigator は以下のポートを使用します。

ポート番号	説明
50000	ソケット



ファイアウォール機能でポート制限をしている場合は、Server Navigator で使用するポートを開放する必要があります。

サポート範囲

以下に Server Navigator のアップデート機能でサポートするデバイスの一覧を表記します。

デバイス	機種		種別	備考
	HA8000	BS500		
BIOS/EFI	OK	-	F/W	アップデート機能は HA8000 のみサポートしています。ダウングレードは未サポートです。
BMC	OK	-	F/W	アップデート機能は HA8000 のみサポートしています。
1G LAN (Broadcom) *1	OK	OK	Driver, Utility	*7
10G LAN (Broadcom) *2	OK	-	F/W, Driver, Utility	*7
8G FC (Hitachi) *3	OK	OK	F/W, Driver, Utility	
LSI 社製 SAS 2008 *4	-	OK	F/W, Driver, Utility	*8
LSI 社製 SAS 2108 *5	OK	-	F/W, Driver, Utility	*8
LSI 社製 SAS 2208 *6	OK	OK	F/W, Driver, Utility	*8
Software RAID	OK	OK	Driver, Utility	*8 *9

*1 HA8000 4 ポート Gigabit LAN ボード
 *1*4 BS500 1Gb 8 ポート イーサネット+RAID(iMR)メザニンカード
 *2 HA8000 I/O ボード LAN 10Gbps x2 ポート
 *3 HA8000 Gigabit Fibre Channel ボード 8Gbps, BS500 Hitachi 8Gb 4 ポート ファイバチャネルメザニンカード, Hitachi 8Gb 2 ポート ファイバチャネルメザニンカード
 *4 BS500 RAID(iMR)メザニンカード
 *5 HA8000 ディスクアレイコントローラボード(外付け用)MegaRAID (LSI SAS2108) SAS 9280-8e, SAS 9267-8i
 *6 HA8000 ディスクアレイコントローラボード(内蔵用) (標準搭載)MegaRAID (LSI SAS2208) SAS 9267-8i, SAS 9266-8i, BS500 RAID 拡張カード
 *7 Broadcom LAN(*1,*2)をアップデートする際は、Provider のアップデートが必要になります。
 LAN のアップデートモジュールのアップデート時に、Provider も自動で更新されますので追加の作業は必要ありません。
 *8 LSI 社製 SAS 2108/SAS2208 用 Utility として HRN と MSM をサポートしております。ただし、HRN と MSM を両方インストールした構成は未サポートです。
 *9 HA8000 では Software RAID の Utility のアップデート機能は、HRN のみサポートしています。

以下に Server Navigator のアップデート機能およびツール連携機能でサポートするアプリケーションの一覧を表記します。

アプリケーション		アップデート	ツール連携
アップデート管理	Hitachi Server Navigator	OK	-
RAID 管理	HRN (Hitachi RAID Navigator)	OK	OK
	MSM (MegaRAID Storage Manager)	OK	-
監視	Log Monitor	OK *	OK
	Alive Monitor	OK *	-
FC 管理	HFCTools	OK	-
	OCM (One Command Manager)	OK	-
LAN/CNA 管理	BACS (Broadcom Advanced Control Suite)	OK	-
	OCM (One Command Manager)	OK	-
	OneCommand NIC Teaming and VLAN Manager	OK	-

* Log Monitor、Alive Monitor のダウングレードは未サポートです。



Server Navigator 自身のアップデートは自動アップデートまたは手動アップデート実行時に先行して行います。また、ロールバックは未サポートです。

制限事項

Server Navigator における制限

ここでは、Server Navigator を使用する前に知っておいていただきたい制限事項を説明します。

- 仮想環境(Hyper-V、Virtage、VMware、等)のゲスト OS での Server Navigator の使用は未サポートです。Virtage の LPAR(ゲスト OS)にインストールしようとした場合は、“This machine is un-support model.”のメッセージを表示して中断します。
- 仮想ドライブを割り当てるため、ドライブ C~Z まで使用している環境では使用できません。
- 100 個を超える多数のディスクドライブがシステム装置に存在する場合には正常に動作しない場合があります。

Server Navigatorのインストール

この章では、Server Navigator のインストール方法およびアンインストール方法について説明します。

- [インストール](#)
- [アップグレード](#)
- [アンインストール](#)



Server Navigator のインストールについて :

- 本章に記載される内容は、Server Navigator の「インストール機能(インストーラ)」を使用しない場合のインストール方法です。
- Server Navigator は、Server Navigator OS インストール機能によるインストールに対応しています。手順は「Hitachi Server Navigator OS セットアップガイド」を参照してください。



Server Navigator のインストールは GUI で操作を行う通常インストールと、操作の必要なサイレントインストールがあります。サイレントインストールはバッチ処理などで自動でインストールを行うのに有効です。アップグレードおよびアンインストールも同様です。

インストール

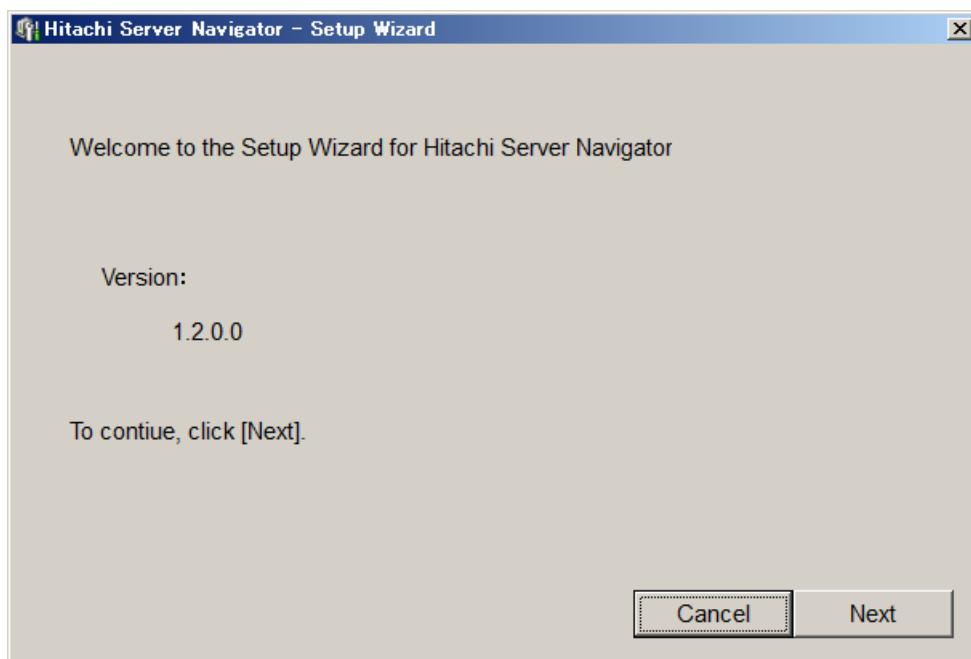
Server Navigator のインストール方法について説明します。

- [Windows版Server Navigatorをインストールする場合 \(P.2-2\)](#)
- [Windows版Server Navigatorをサイレントインストールする場合 \(P.2-6\)](#)

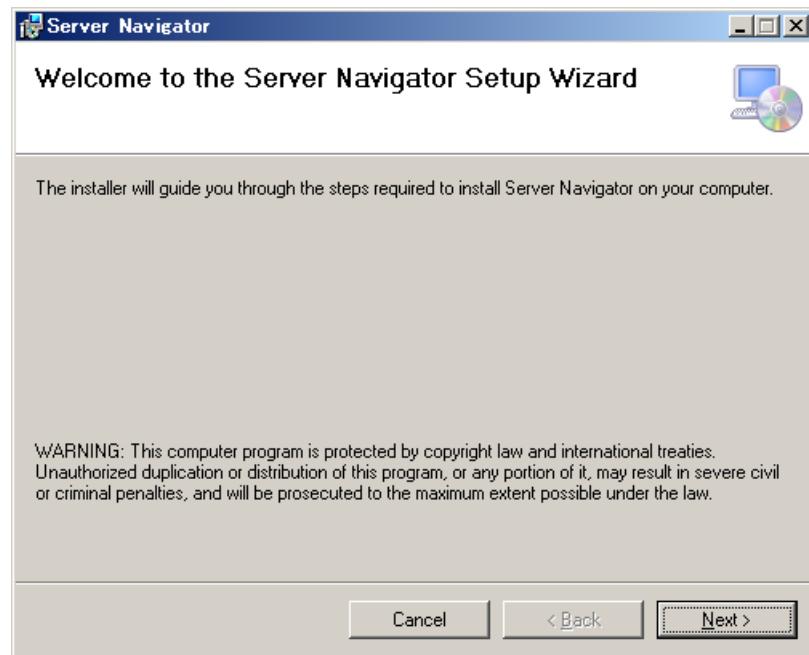
Windows版Server Navigatorをインストールする場合

Windows 版 Server Navigator のインストール方法について説明します。

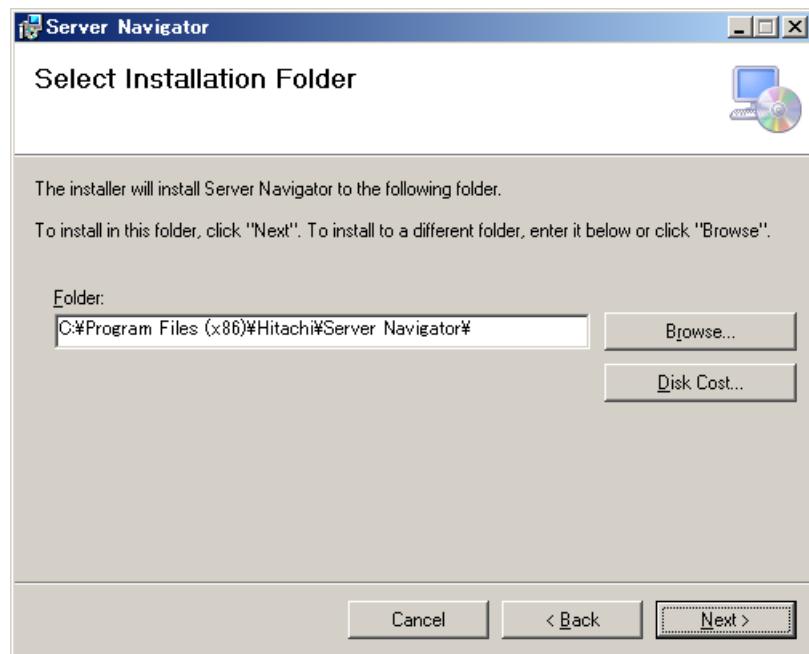
1. Windows を立ち上げ、"Administrator" でログオンします。
2. CD/DVD ドライブに "Hitachi Server Navigator DVD" を入れます。
3. 次のフォルダに移動します。
"d:¥WinSrv2008¥Utility¥SNV"
d: は CD/DVD ドライブです。ご使用の環境に置き換えてください。
4. setup.exe が表示されたら 「**setup.exe**」 をダブルクリックして実行します。
5. 次の画面が表示されたら 「**Next**」 ボタンをクリックします。



6. "Welcome to the Server Navigator Setup Wizard" 画面が表示されたら「Next」ボタンをクリックします。



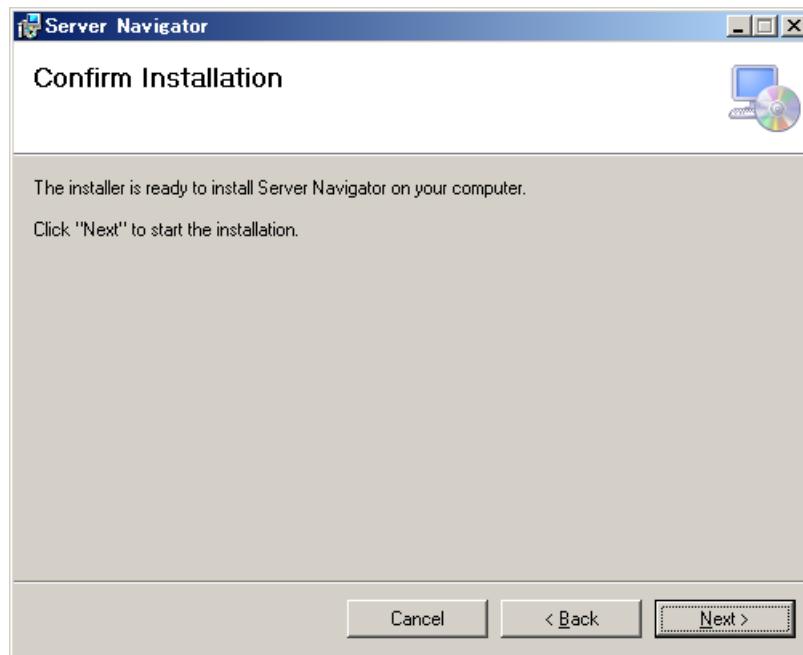
7. "Select Installation Folder" 画面が表示されたら、インストールフォルダを選択し、「Next」ボタンをクリックします。



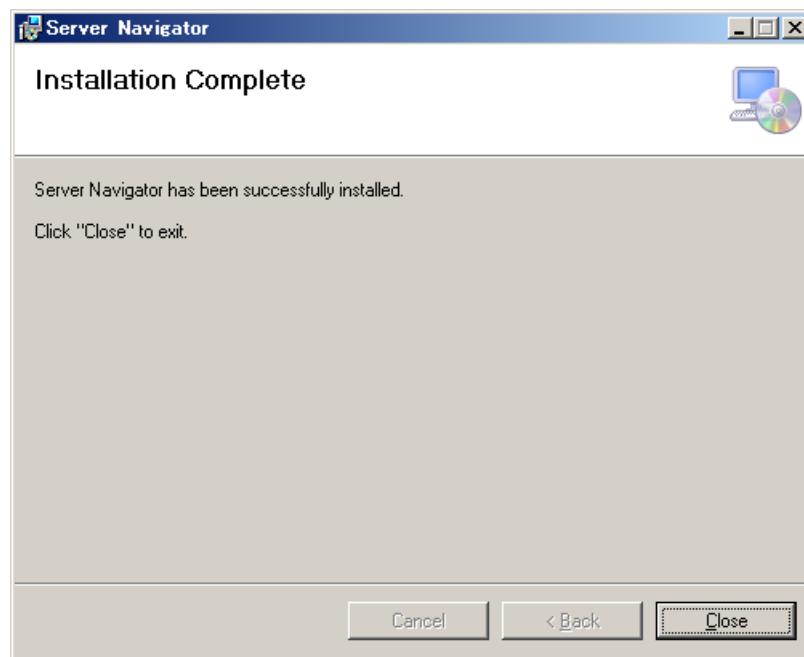


標準インストール先ディレクトリは、32 ビット版 OS では C:\Program Files\Hitachi\Server Navigator、64 ビット版 OS では C:\Program Files (x86)\Hitachi\Server Navigator です。
本ユーティリティは 32 ビットアプリケーションであり、64 ビット版 OS で C:\Program Files\ 以下にインストールすることはできません。

8. "Confirm Installation" の画面が表示されたら「Next」ボタンをクリックします。



9. インストールが完了すると "Installation Complete" 画面が表示されますので、「Close」ボタンをクリックします。



10. 次の画面が表示されるので DVD を取り出し、「OK」ボタンをクリックします。



11. 次の画面が表示されるので「はい」ボタンをクリックします。すぐにシステム装置が再起動します。



12. Hitachi Server Navigator がインストールされると、デスクトップに下記ショートカットが作成されます。



以上で Hitachi Server Navigator のインストールは終了です。

Windows版Server Navigatorをサイレントインストールする場合

Windows 版 Server Navigator のサイレントインストール方法について説明します。

1. Windows を立ち上げ、"Administrator" でログオンします。
2. CD/DVD ドライブに "Hitachi Server Navigator DVD" を入れます。
3. コマンドプロンプトから次のコマンドを入力して、SNV フォルダに移動してください。
cd d:¥WinSrv2008¥Utility¥SNV
d:は CD/DVD ドライブです。ご使用の環境に置き換えてください。
4. コマンドプロンプトから下記コマンドを入力して、Hitachi Server Navigator をサイレントインストールします。
snvsetup.exe -install
5. Hitachi Server Navigator は下記のフォルダにインストールされます。
 - 32 ビット版 OS : C:¥Program Files¥Hitachi¥Server Navigator
 - 64 ビット版 OS : C:¥Program Files (x86)¥Hitachi¥Server Navigator

「The installation was successful. (Please eject the CDROM.)」が表示されると、Hitachi Server Navigator のインストールは終了です。DVD を取り出してください。

```
D:¥>cd d:¥WinSrv2008¥Utility¥SNV  
D:¥>snvsetup.exe -install  
The installation was successful. (Please eject the CDROM.)  
D:¥>
```



Hitachi Server Navigator の構成を有効にするために、インストール後は必ずシステムを再起動してください。

アップグレード

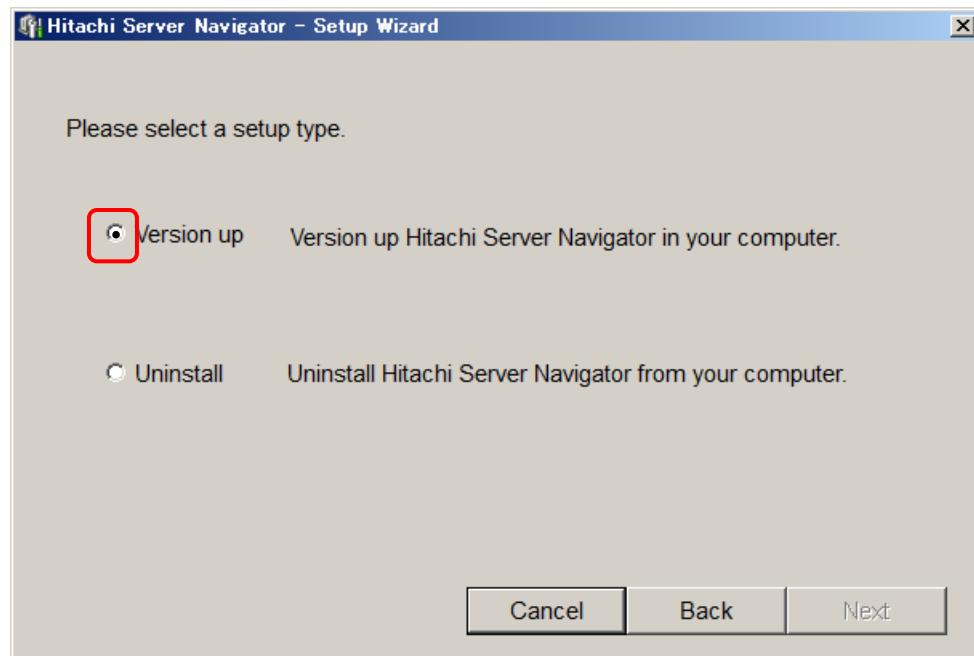
Server Navigator のアップグレード方法について説明します。

- [Windows版Server Navigatorをアップグレードする場合 \(P.2-7\)](#)
- [Windows版Server Navigatorをサイレントインストールでアップグレードする場合 \(P.2-8\)](#)

Windows版Server Navigatorをアップグレードする場合

Windows版Hitachi Server Navigatorのアップグレードは、「[Windows版Server Navigatorをインストールする場合](#)」(P.2-2)と同様の手順を実行してください。

ただし、手順 3 の後に次の画面が表示されたら「**Version up**」を選択し、「**Next**」ボタンをクリックしてください。



また、手順 7 の後に次の画面が表示されたら「**OK**」ボタンをクリックし、以降のインストール手順を引き続き実行してください。



以上で Hitachi Server Navigator のアップグレードは終了です。

Windows版Server Navigatorをサイレントインストールでアップグレードする場合

「[Windows版Server Navigatorをサイレントインストールする場合](#)」(P.2-6)と同様の手順で、サイレントインストールします。



Hitachi Server Navigator のダウングレードはできません。

アンインストール

Server Navigator のアンインストール方法について説明します。

- [Windows版Server Navigatorをアンインストールする場合 \(P.2-10\)](#)
- [Windows版Server Navigatorをサイレントアンインストールする場合 \(P.2-13\)](#)



Windows の「プログラムと機能」では正常にアンインストールできません。必ず、"Hitachi Server Navigator DVD" を使用したアンインストールを実行してください。



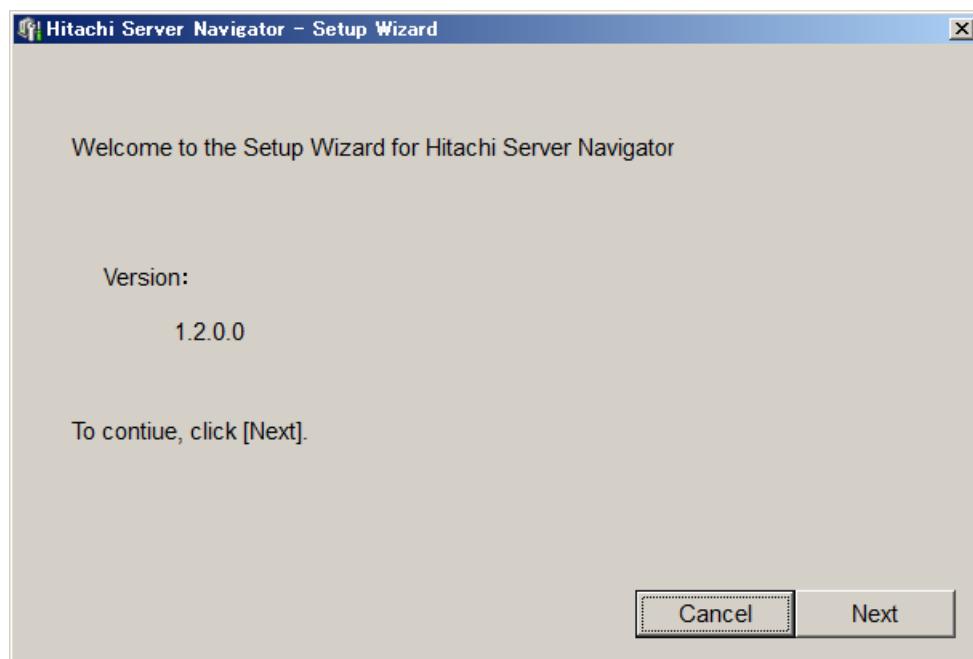
アンインストール作業前に、Server Navigator GUI の 「[Versionボタン](#)」 (P.3-4) で導入しているVersionを確認してください。

Server Navigator Ver.1.1.0.x(Hitachi Server Navigator CD-ROM 01-0x)のメディアでは、Server Navigator Ver.1.2.0.x はアンインストールできません。その場合は、日立アップデートサイトより、Server Navigator Ver.1.2.0.x を入手してアンインストールを行ってください。

Windows版Server Navigatorをアンインストールする場合

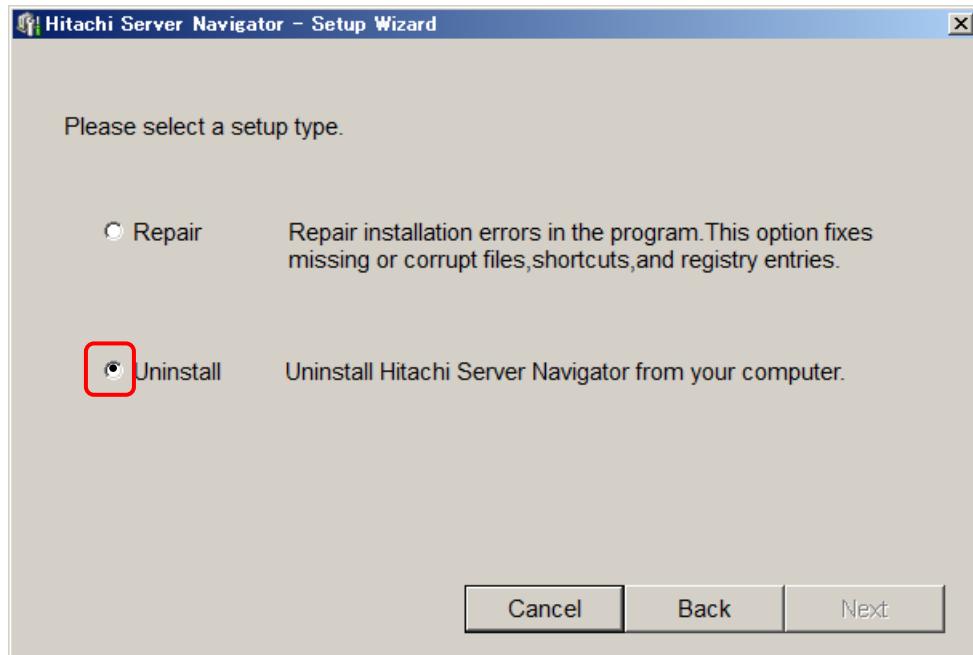
Windows 版 Server Navigator のアンインストール方法について説明します。

1. Windows を立ち上げ、"Administrator" でログオンします。
2. CD/DVD ドライブに "Hitachi Server Navigator DVD" を入れます。
3. 次のフォルダに移動します。
"d: ¥WinSrv2008¥Utility¥SNV"
d: は CD/DVD ドライブです。ご使用の環境に置き換えてください。
4. setup.exe が表示されたら「**setup.exe**」をダブルクリックして実行します。
5. 次の画面が表示されたら「**Next**」ボタンをクリックします。

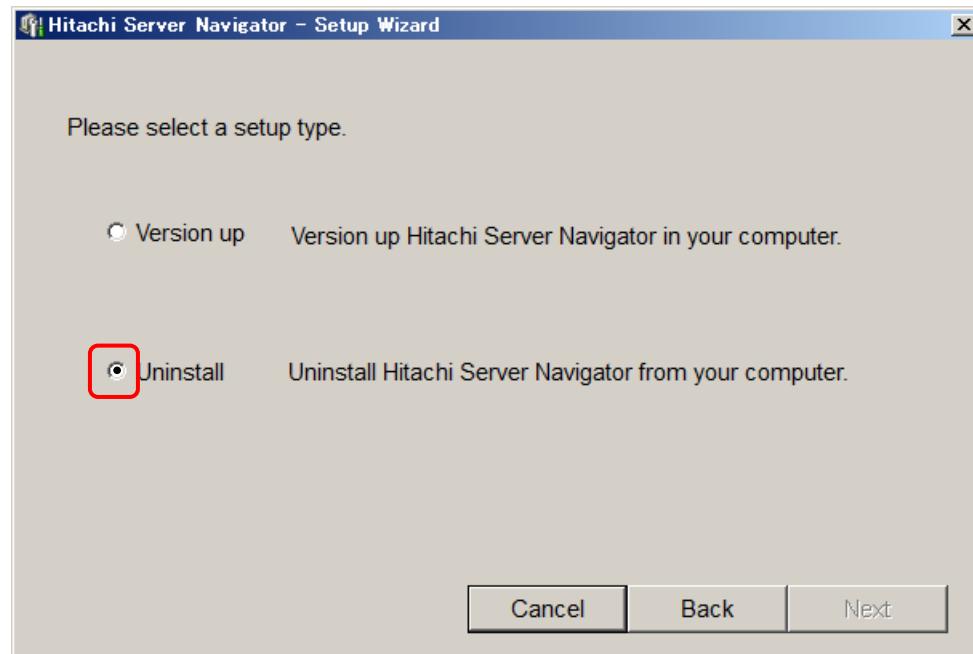


6. 「Please select a setup type.」の画面が表示されたら「Uninstall」を選択し、「Next」ボタンをクリックします。なお、画面は状況により異なります。

- インストールバージョンと同じバージョンの "Hitachi Server Navigator DVD" を使用した場合



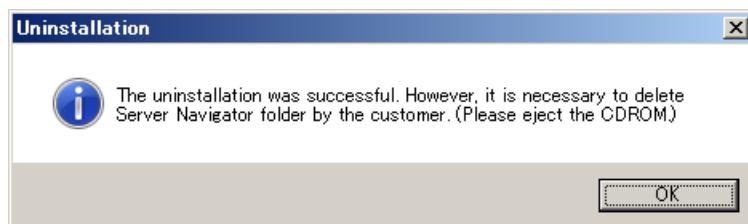
- インストールバージョンより新しいバージョンの "Hitachi Server Navigator DVD" を使用した場合



7. 次の画面が表示されたら「OK」ボタンをクリックします。



8. 次の画面が表示されたら DVD を取り出し、「OK」ボタンをクリックします。



9. 次の画面が表示されるので「はい」ボタンをクリックします。すぐにシステム装置が再起動します。



以上で Hitachi Server Navigator のアンインストールは終了です。

Windows版Server Navigatorをサイレントアンインストールする場合

Windows 版 Server Navigator のサイレントアンインストール方法について説明します。

1. Windows を立ち上げ、"Administrator" でログオンします。
2. CD/DVD ドライブに "Hitachi Server Navigator DVD" を入れます。
3. コマンドプロンプトから次のコマンドを入力して、SNV フォルダに移動してください。
cd d:¥WinSrv2008¥Utility¥SNV
d:は CD/DVD ドライブです。ご使用の環境に置き換えてください。
4. コマンドプロンプトから下記コマンドを入力して、Hitachi Server Navigator をサイレントアンインストールします。
snvsetup.exe -uninstall

「The uninstallation was successful. (Please eject the CDROM.)」が表示されると、Hitachi Server Navigator のアンインストールは終了です。DVD を取り出してください。

```
D:¥>cd d:¥WinSrv2008¥Utility¥SNV  
D:¥>snvsetup.exe -uninstall  
The uninstallation was successful. (Please eject the CDROM.)  
D:¥>
```

5. システムを再起動してください。

Server NavigatorのGUI画面の説明

この章では、Server Navigator の GUI 画面について説明します。

- [Server Navigator GUIの起動](#)
- [アップデート機能用GUI画面](#)
- [監視機能設定機能用GUI画面](#)
- [ツール連携機能用GUI画面](#)

Server Navigator GUIの起動

Server Navigator GUI を起動するには、デスクトップ上に作成された「Hitachi Server Navigator」のショートカットを実行してください。



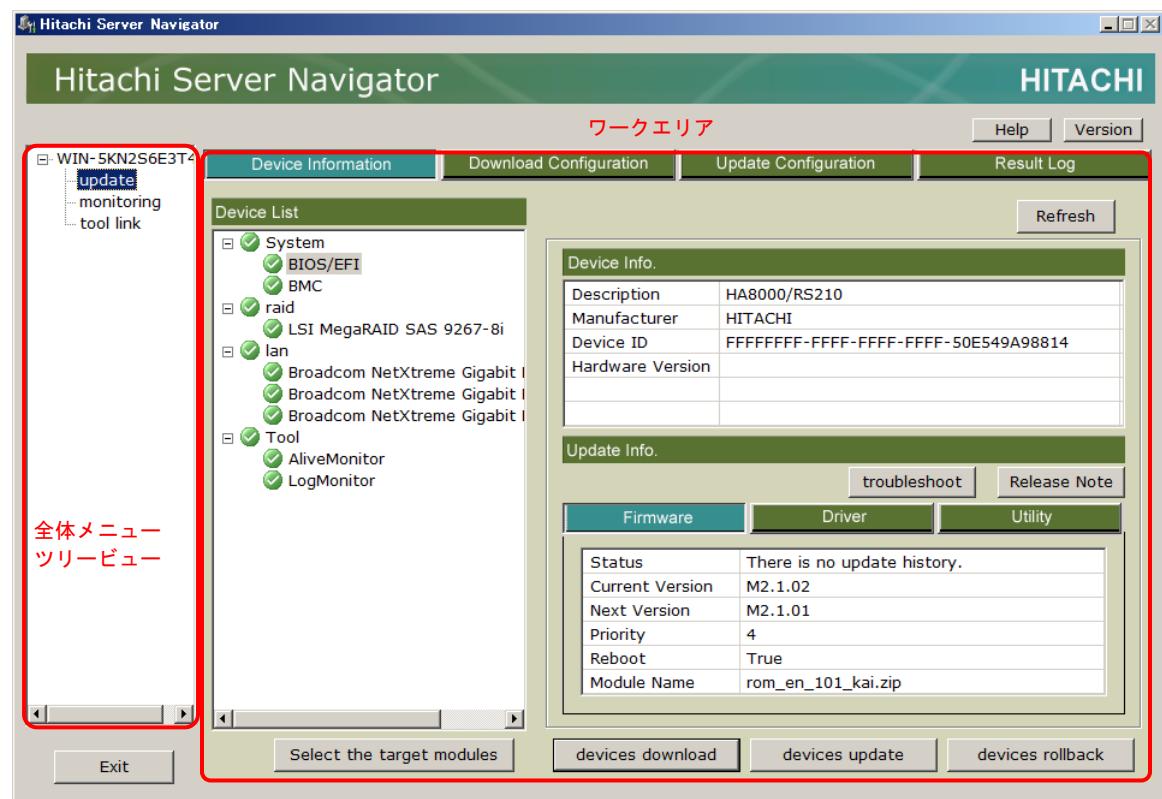
アップデートを安定的に行うため、自動アップデート動作中は、Server Navigator GUI を起動しないでください。

Server Navigator GUI のトップ画面の共通部について説明します。

全体メニューによる各機能の切り替え、「Hitachi Server Navigator ユーザーズガイド」(本ドキュメント)の表示、本ユーティリティのバージョン表示、GUI 画面の終了を行うことが可能です。

トップ画面共通部

Server Navigator GUI 起動時の初期画面である、アップデート機能用 GUI 画面のデバイス情報画面を例にトップ画面の共通部について詳細を説明します。



全体メニューツリービュー

コンピュータ名称と Hitachi Server Navigator の各機能(update、monitoring、tool link)がメニュー表示されます。メニューに表示された機能の名称をクリックすることで、機能を切り替えることが可能です。Server Navigator GUI 起動時は update 機能がデフォルトで選択されます。

- Update メニュー

[アップデート機能用GUI画面](#)に切り替えます。

- Monitoring メニュー

[監視機能設定機能用GUI画面](#)に切り替えます。

- tool link メニュー

[ツール連携機能用GUI画面](#)に切り替えます。

ワークエリア

全体メニューツリービューで選択された機能に対応する内容が表示されます。

Exit ボタン

Server Navigator GUI を終了します。「**ESC**」キーで代替可能です。

確認ダイアログが表示され、「はい」を選択することにより、Server Navigator GUI を終了します。「いいえ」を選択すると元の画面に戻ります。



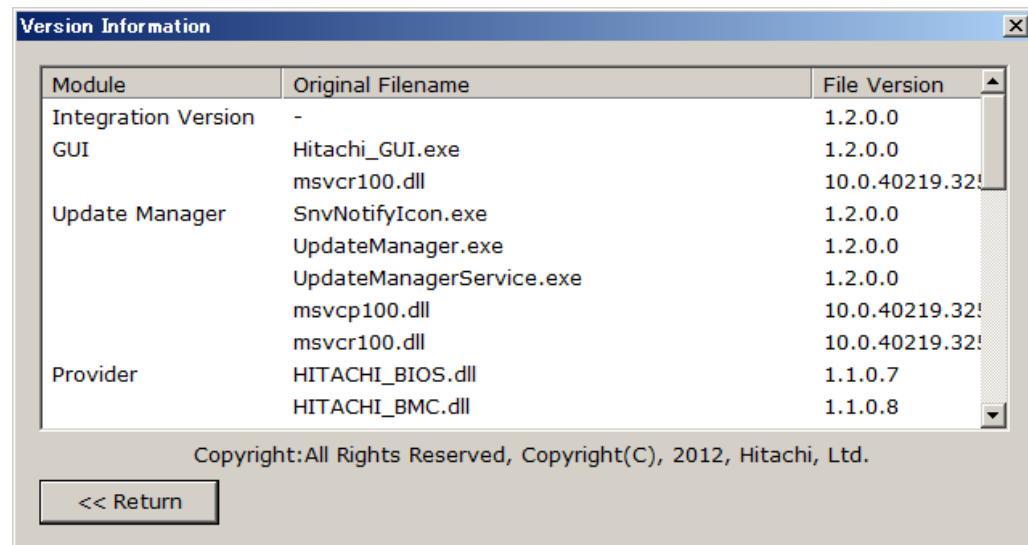
Help ボタン

「Hitachi Server Navigator ユーザーズガイド」(本ドキュメント)を Adobe Reader を使用して表示します。

Version ボタン

バージョン情報ダイアログ画面を表示します。

本ダイアログ画面に Server Navigator のバージョン情報を表示します。「Return」ボタンまたは「ESC」キーで元の画面に戻ります。



統合バージョンおよび構成するファイル(exe,dll)のバージョン情報を取得して表示します。
Return ボタンをクリックすることにより、本画面を閉じて前の画面に戻ります。

項目	説明
Module	Integration Version(統合バージョン)、"Hitachi Server Navigator"を構成するモジュール(GUI/Update Manager/CIM-Provider)を表示します。
Original Filename	Module を構成するファイル名を昇順で表示します。
File Version	ファイルバージョンを表示します。



Server Navigator のバージョン表示について:

- Server Navigator のバージョンは、バージョン情報ダイアログ画面の「Integration Version」の"File Version"の値により確認できます。
- OS の「コントロールパネル」 – 「プログラムと機能」で表示されるバージョンは、インストール時のバージョンです。Server Navigator 自身のアップデートが行われた場合、バージョン情報ダイアログ画面で表示されるバージョン(正しいバージョン)と不一致となります。

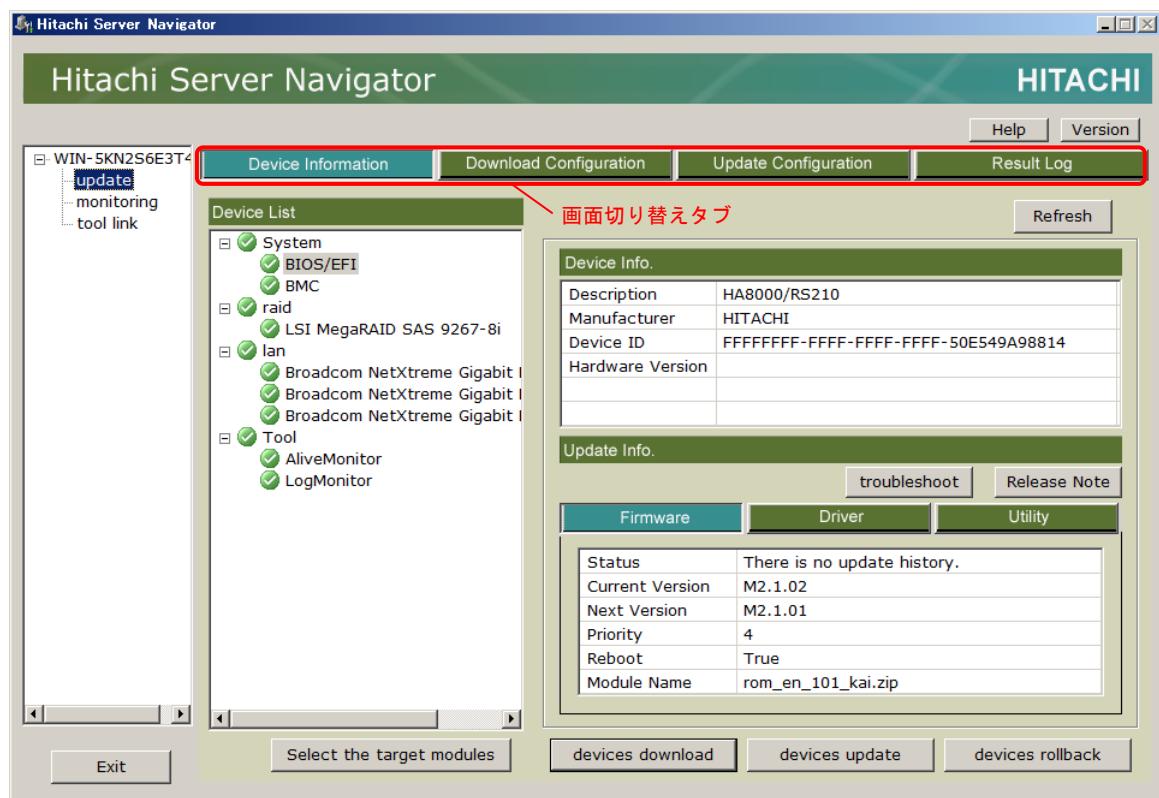
アップデート機能用GUI画面

全体メニュー「ツリービュー」の「update」メニューに対応する、アップデート機能用の GUI 画面です。

Server Navigator でサポートするデバイスの F/W、driver、utility およびツールのバージョン管理やアップデート管理が可能です。

アップデート機能用GUI画面共通部

アップデート機能用 GUI 画面の共通部について詳細を説明します。



画面切り替えタブ

Server Navigator のアップデート機能における、各画面の切り替えを行うためのタブです。クリックすることにより、アップデート機能の対応する画面に切り替えます。

- Device Information タブ

[デバイス情報画面](#)に切り替えます。

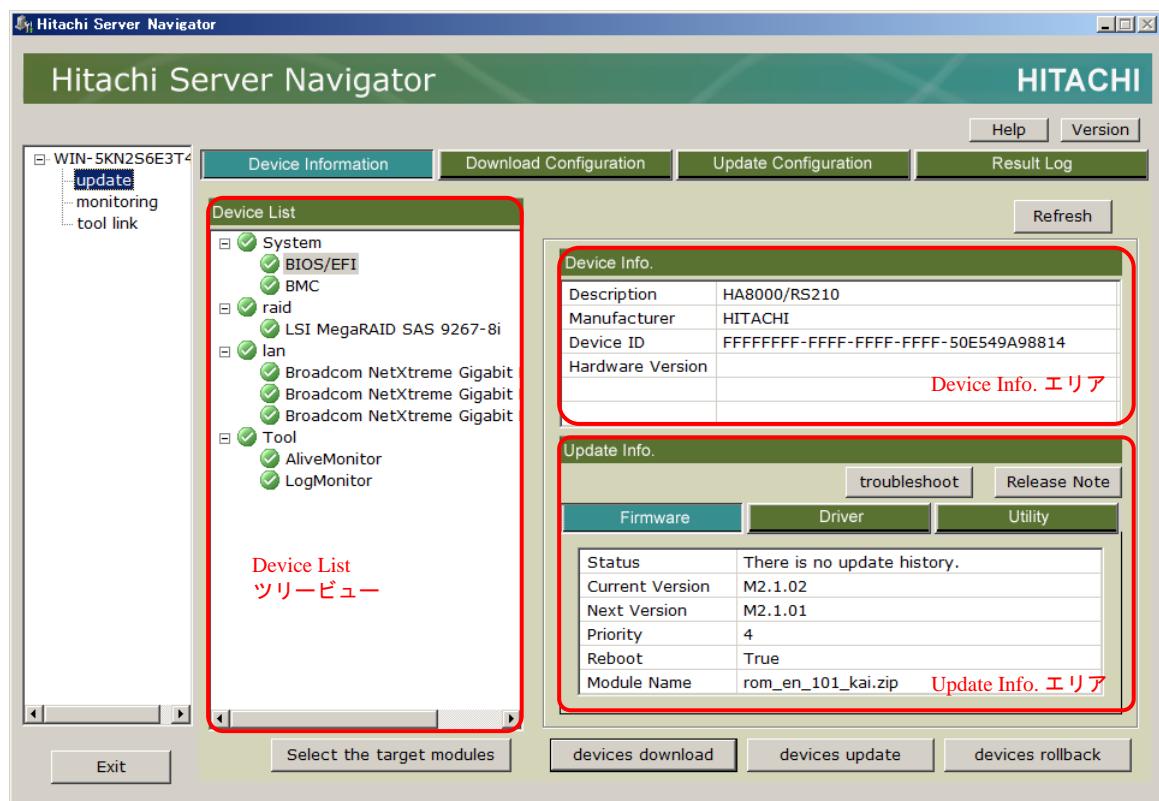
- Download Configuration タブ
[ダウンロード設定画面](#)に切り替えます。
- Update Configuration タブ
[アップデート設定画面](#)に切り替えます。
- Result Log タブ
[トレースログ画面](#)に切り替えます。

デバイス情報画面

Server Navigator GUI を起動すると、アップデート機能のデバイス情報画面を表示します。全体メニュー ツリービューの「update」メニュー選択後、Device Information タブに対応している画面です。

サーバに接続されたデバイスの一覧およびバージョンの確認と、アップデートモジュールの情報の確認を行うことが可能です。また、ダウンロード、アップデート、ロールバックの開始を行うことが可能です。

デバイス情報画面について詳細を説明します。



Device List ツリービュー

接続されているデバイスの一覧をツリービューで表示します。また、各デバイスの状態をアイコンで表示します。Device Info. エリアおよび Update Info. エリアに情報を表示するデバイスの選択を行います。



FC(Hitachi PCI Fibre Channel Adapter)をご使用のシステム構成で、本ユティリティを使用する場合は、HFCTools のインストールが必要です。インストールしていない場合は、FC の状態表示およびアップデートを正常に行うことができません。

以下にアイコンとデバイスの状態の関係を表記します。

アイコン	デバイスの状態
	Select the target modules 画面で選択されたデバイス、かつ、アップデート情報がないデバイスを表します。
	Select the target modules 画面で選択されたデバイス、かつ、アップデート情報があるデバイスを表します。
—	Select the target modules 画面で未選択のデバイスを表します。

Refresh ボタン

Device Info. エリアのデバイス詳細情報を各デバイス管理部から再取得し表示します。
「ENTER」キーで代替可能です。

Device Info. エリア

OS から取得したデバイスの詳細情報を表示します。

項目	説明
Description	デバイスの詳細説明情報を表示します。
Manufacturer	デバイスの製造元情報を表示します。
Device ID	デバイスを識別する文字列を表示します。
Hardware Version	ハードウェアのバージョンを表示します。 デバイスプロバイダが未サポートのデバイスについては、"N/A"と表示します。
補足情報	デバイスに依存する補足情報を表示します。
BIOS/EFI	補足情報はありません。(空白表示します。)
BMC	補足情報はありません。(空白表示します。)
RAID	コントローラ番号を表示します。
LAN	MAC アドレスを表示します。
FC	WWN アドレスを表示します。

Update Info. エリア

Server Navigator の各デバイス管理部から取得したアップデート関連の情報を表示します。
ファームウェア、ドライバ、ユーティリティに表示内容を切り替えるタブと、リカバリマニュアルとリリースノート表示するボタンを備えます。

項目	説明
FW タブ	デバイスのファームウェアの情報に切り替えます。
Driver タブ	デバイスのドライバの情報に切り替えます。
Utility タブ	デバイスのユーティリティの情報に切り替えます。

項目	説明
troubleshoot ボタン	リカバリマニュアルを表示します。
Release Note ボタン	リリースノートを表示します。
Status	更新モジュールの適用状態を表示します。
Module Name	適用モジュールの物理ファイル名称を表示します。
Current Version	現在稼動しているベンダバージョンを表示します。稼動バージョンとも表記します。
Next Version	アップデート予定のベンダバージョンを表示します。以下の条件で空白表示します。 - Server Navigator をインストール後、ダウンロードを一回も行っていない場合 - ダウンロードを行ったが、ダウンロードパスが間違っていた場合 - 更新重要度がダウンロードを許可する重要度より低くダウンロードしなかった場合 - 稼動バージョンと同じバージョンをダウンロードした場合 - アップデートまたはロールバックが成功した場合
Priority	更新重要度(1~5)を表示します。 1 : 軽微。マイナーな問題の修正を意味します。 2 : 注意。最先の便宜を図ってインストールしてください。影響力が中または小の問題の修正を意味します。 3 : 重要。最先の便宜を図ってインストールしてください。影響力が高い問題の修正を意味します。 4 : 必須。可能な限り早くインストールしてください。 5 : (reserved)
Reboot	アップデート後にリブートが必要か否かを表示します。 - True : 必要 - False : 不要

以下に Status の一覧を表記します。

Status	意味
"There is no update history."	アップデート履歴なし。アイドル状態
"Executing backup"	バックアップ中
"Backup completion"	バックアップ完了
"Backup failure" (文字色赤)	バックアップ失敗
"Executing update preprocessing"	アップデート前処理中
"Waiting for reboot.(Update preprocessing)"	アップデート前処理完了／リブート待ち
"Update preprocessing completion."	アップデート前処理完了
"Update preprocessing failure" (文字色赤)	アップデート前処理失敗
"Executing update"	アップデート中
"Waiting for reboot.(Update)"	アップデート完了／リブート待ち
"Update completion"	アップデート完了

Status	意味
"Update failure" (文字色赤)	アップデート失敗
"Executing restore"	リストア中
"Waiting for reboot.(Restore)"	リストア完了／リブート待ち
"Restore completion"	リストア完了
"Restore failure" (文字色赤)	リストア失敗
"Executing an application"	適用判定中
"Application success"	適用成功
"Application failure" (文字色赤)	適用失敗

troubleshoot ボタン

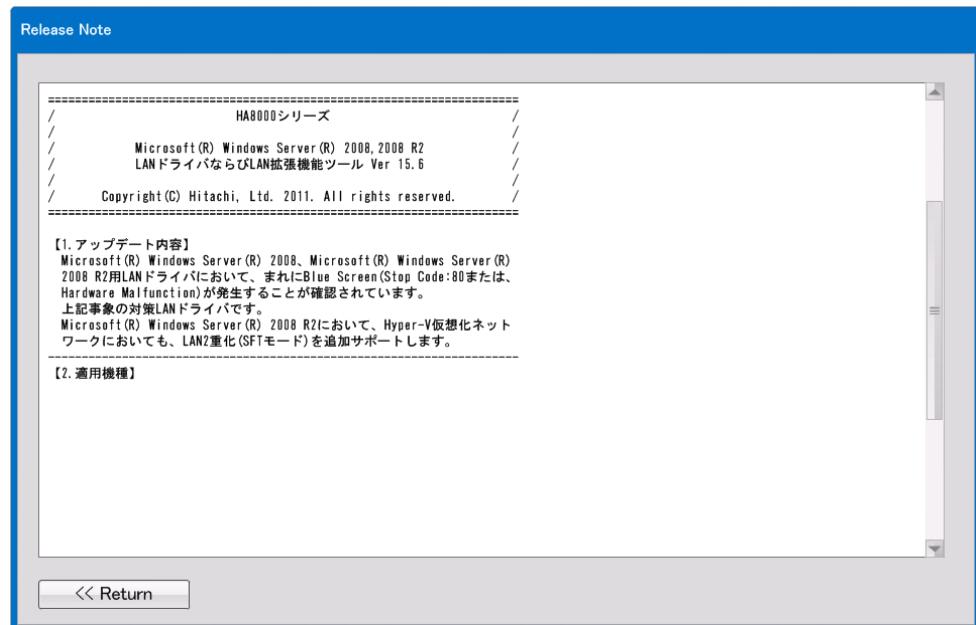
Device Info. エリアに表示中のアップデートモジュールに対応するリカバリマニュアルを Adobe Reader を使用して表示します。

対応するマニュアルが存在しない場合は、本ボタンは不活性状態となります。

Release Note ボタン

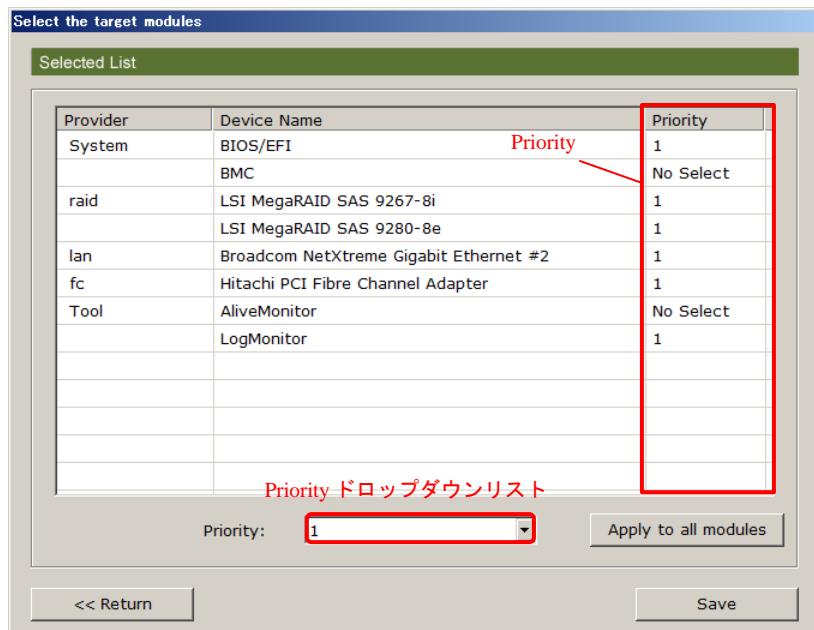
Device Info. エリアに表示中のアップデートモジュールに対応するリリースノートダイアログ画面を表示します。Return ボタンをクリックすることにより、デバイス情報画面に戻ります。

対応するリリースノートが存在しない場合は、本ボタンは不活性状態となります。



Select the target modules ボタン

Select the target modules ダイアログ画面(ターゲットモジュール選択画面)を開きます。デバイス情報画面の Device List ツリービューの表示情報をもとにデバイス一覧とデバイスの適用重要度を表示します。管理対象コンポーネントの適用重要度を変更することにより、**devices download** ボタン/**devices Update** ボタン/**devices rollback** ボタンをクリックしたときのダウンロード/アップデート/ロールバックを行うコンポーネントの選択を行います。以降、ターゲットモジュール選択機能とです。



以下に Select the target modules 画面の表示項目を表記します。

項目	説明
Provider	プロバイダ名称を表示します。
Device Name	デバイス名を表示します。
Priority	デバイスの適用重要度を表示します。 "No Select" (適用なし)、および、1~5 の候補リストから設定可能です。アップデートモジュールの重要度(GUI画面のDevice InformationタブのUpdate Info. エリアの「Priority」(P.3-9)参照)の値が本設定値以上の場合、ダウンロード/アップデート/ロールバック対象となります。 "No Select" : すべて対象外 1 : すべてのアップデートモジュールが対象 2 : 重要度が 2 以上のアップデートモジュールが対象 3 : 重要度が 3 以上のアップデートモジュールが対象 4 : 重要度が 4 以上のアップデートモジュールが対象 5 : 重要度が 5 のアップデートモジュールが対象
Priority ドロップダウンリスト	一括設定時のデバイスの適用重要度を表示します。

項目	説明
Apply to all modules ボタン	Priority ドロップダウンリストで選択されている適用重要度を一括で反映します。
Save ボタン	選択状態および Priority を保存します。
Return ボタン	本画面を閉じて、前画面に戻ります。

devices download ボタン

Select the target modules 画面で選択されたコンポーネントのアップデートモジュールを指定の方法でダウンロードします。

Select Download Mode 画面が開きますので、ダウンロード接続先を選択します。



項目	説明
From Media Drive.ラジオボタン	アクティブになると、ドライブ指定ダウンロードパステキストボックスに設定されたパスをダウンロード接続先として選択します。 From Web.ラジオボタンと二者選一でアクティブに設定可能で、初期状態では From Media Drive.ラジオボタンがアクティブです。
From Web or Cache Server.ラジオボタン	アクティブになると、ダウンロード設定で設定した Web サイトまたはキャッシュサーバをダウンロード接続先として選択します。
ドライブ指定ダウンロードパステキストボックス	手動ダウンロードで使用するダウンロード元のドライブ、パスを指定します。 「Hitachi Server Navigator ユーザーズガイド アップデート機能」の「ダウンロードユーティリティー」で作成したフォルダを指定してください。 最大入力桁数:128 桁(半角全角混在)
Refer ボタン	フォルダ参照ダイアログが表示されます。
Download ボタン	手動ダウンロードを実行します。
Return ボタン	本画面を閉じて、前の画面に戻ります。

Refer ボタン

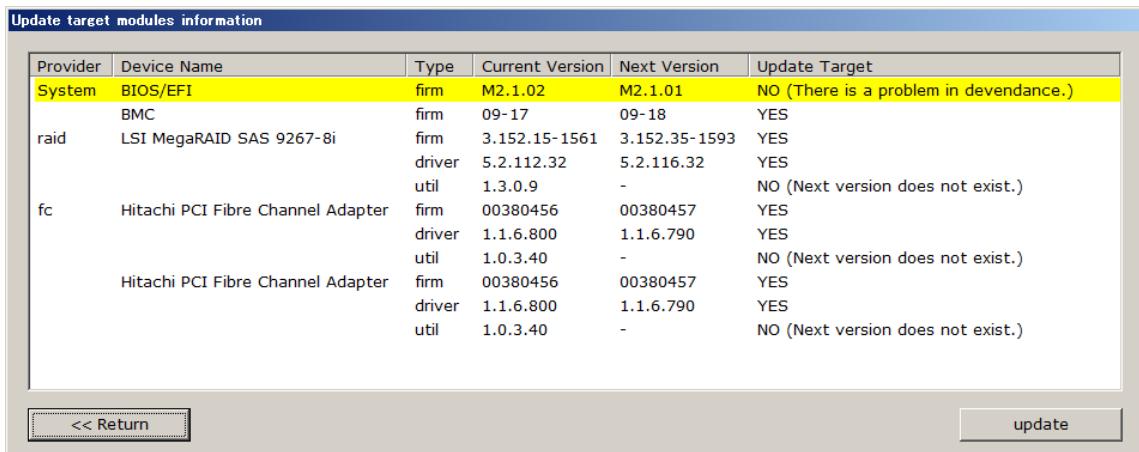
フォルダ参照ダイアログが表示されます。選択したパスがドライブ指定ダウンロードパステキストボックスに入力されます。



devices Update ボタン

Select the target modules 画面で選択されたコンポーネントの手動アップデートを開始するためのボタンです。

Update target modules information 画面が開きますので、アップデートを実行するか、中止するかを選択します。



項目 *	説明
Provider リストビュー	プロバイダ名を表示します。
Device Name リストビュー	デバイス名を表示します。
Type リストビュー	モジュール種別(Firmware/Driver/Utility)を表示します。
Current Version リストビュー	現在稼働しているベンダバージョンを表示します。
Next Version リストビュー	アップデート予定のベンダバージョンを表示します。
Update Target リストビュー	アップデート情報がない場合は"- "を表示します。 依存関係チェックの結果を評価し、アップデートが出来るかどうかの情報を表示します。

項目 *	説明
update ボタン	Update Target リストビューで"YES"となっているコンポーネントの手動アップデートを実行します。
Return ボタン	本画面を閉じて、前の画面に戻ります。
* 背景が赤色の場合は、エラーが発生していてアップデートできないことを示しています。	
背景が黄色の場合は、依存関係を満たしておらずアップデートできないことを示しています。	



システム装置やアップデート対象となるコンポーネントに障害が発生していると、正常にアップデートすることはできません。エラーランプの確認やOSイベントログの確認等の障害検知を事前に行ってください。障害検知の方法については、システム装置の製品ユーザーズガイドを参照してください。

devices rollback ボタン

Select the target modules 画面で選択されたコンポーネントの手動ロールバックを開始するためのボタンです。一つ前のアップデート適用前の稼動バージョン用のアップデートモジュールを適用します。

Update target modules information 画面が開きますので、アップデートを実行するか、中止するかを選択します。画面と操作は **devices Update** ボタンと同様です。アップデートをロールバック、**update** ボタンを **rollback** ボタンに置き換えてください。



Update target modules information 画面と操作の説明は、**devices Update** ボタンと同様です。アップデートをロールバックに置き換えて参照してください。



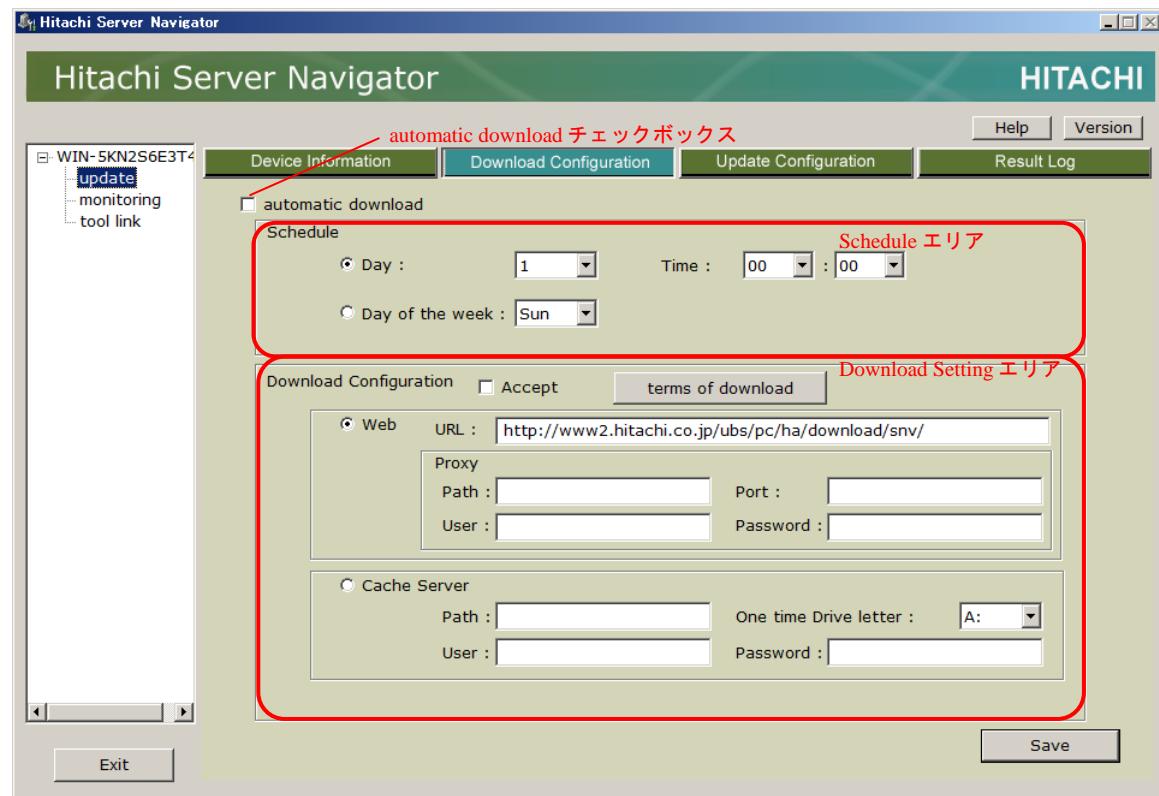
システム装置やロールバック対象となるコンポーネントに障害が発生していると、正常にロールバックすることはできません。エラーランプの確認やOSイベントログの確認等の障害検知を事前に行ってください。障害検知の方法については、システム装置の製品ユーザーズガイドを参照してください。

ダウンロード設定画面

全体メニュー「ツリービュー」の「update」メニュー選択後、Download Configuration タブに対応している画面です。

自動ダウンロードの有効化と、自動ダウンロードのスケジュール設定が可能です。また、自動または手動ダウンロードを行うための接続設定が設定可能です。

ダウンロード設定画面について詳細を説明します。



automatic download チェックボックス

自動ダウンロードの可否を設定します。チェックが入っている状態では、スケジュールに従い自動ダウンロードを開始します。初期状態はチェックが入っていない状態です。



スケジュールに従い自動ダウンロードが行われた場合、automatic download チェックボックスのチェックは解除され、以後自動ダウンロードは行いません。再度自動実行が必要な場合は、自動ダウンロード設定を再設定してください。

Schedule エリア

以下の項目のダウンロードスケジュールを表示および設定します。

項目	説明
Day ラジオボタン	アクティブにすると日付指定でのスケジュールが有効となります。 Day ドロップダウンリストの設定が有効となります。 Day of the week ラジオボタンと二者択一でアクティブに設定可能で、初期状態では Day ラジオボタンがアクティブです。
Day of the week ラジオボタン	アクティブにすると曜日指定でのスケジュールが有効となります。 Day of the week ドロップダウンリストの設定が有効となります。
Day ドロップダウンリスト	日付を指定します。翌日から一ヶ月以内の設定となります。 未選択、01~31 の候補リストから設定可能です。 入力日が当日以下の場合、翌月の設定となります。31 日が存在しない月やうるう年を考慮して存在しない日付の場合、その翌月の 1 日の設定となります。
Day of the week ドロップダウンリスト	曜日を指定します。当日から一週間以内の設定となります。 未選択、Sun~Sat の候補リストから設定可能です。
Hour ドロップダウンリスト	時刻(時)を指定します。(24 時間指定) *
Minutes ドロップダウンリスト	時刻(分)を指定します。 *

* 時刻設定は、各ダウンロードスケジュールで共通です。



自動ダウンロードの開始時刻は、現在時刻から 1 時間以内に設定できません。

Download Setting エリア

以下の項目のダウンロード設定を表示および設定します。

項目	説明		
Accept チェックボックス	ダウンロードに関する使用条件に同意を指定します。 初期状態はチェックが入っていない状態です。		
terms of download ボタン	使用許諾画面を表示します。		
Web エリア	Web ラジオボタン	Web サーバからダウンロードを行う設定になります。	
	URL テキストボックス	URL を指定します。初期値は " http://www2.hitachi.co.jp/ubs/pc/ha/download/snv/ " です。 (Server Navigator 対応日立 Web サイトです。なお、本 URL を直接アクセスすることはできません。) 最大入力桁数: 128 桁(半角英数記号)	
	Proxy エリア	Path テキストボックス	プロキシサーバのアドレスを指定します。初期値は ""(空)です。 最大入力桁数: 128 桁(半角英数記号)
		Port テキストボックス	プロキシサーバのポート番号を指定します。初期値は ""(空)です。 最大入力桁数: 5 桁(半角数字)
		User テキストボックス	プロキシサーバに接続するためのユーザ名を指定します。初期値は ""(空)です。 最大入力桁数: 32 桁(半角英数記号)
		Pass テキストボックス	プロキシサーバに接続するためのパスワードを指定します。ブラインドキャラクタで表示します。初期値は ""(空)です。 最大入力桁数: 32 桁(半角英数記号)
Cache Server エリア	Cache Server ラジオボタン	キャッシュサーバからダウンロードを行う設定になります。	
	Path テキストボックス	キャッシュサーバのアドレスを指定します。初期値は ""(空)です。 最大入力桁数: 128 桁(半角英数記号)	
	One time Drive letter ドロップダ ウンリスト	一時使用ドライブレターを指定します。未選択、A:～Z:の候補リストから設定します。初期値は Z: です。	
	User テキストボックス	キャッシュサーバに接続するためのユーザ名を指定します。初期値は ""(空)です。 最大入力桁数: 32 桁(半角英数記号)	
	Pass テキストボックス	キャッシュサーバに接続するためのパスワードを指定します。ブラインドキャラクタで表示します。初期値は ""(空)です。 最大入力桁数: 32 桁(半角英数記号)	

Save ボタン

ダウンロード設定を保存します。「ENTER」キーで代替可能です。

確認ダイアログが表示され、「はい」を選択することにより、ダウンロード設定が保存されます。「いいえ」を選択するとダウンロード設定画面に戻ります。

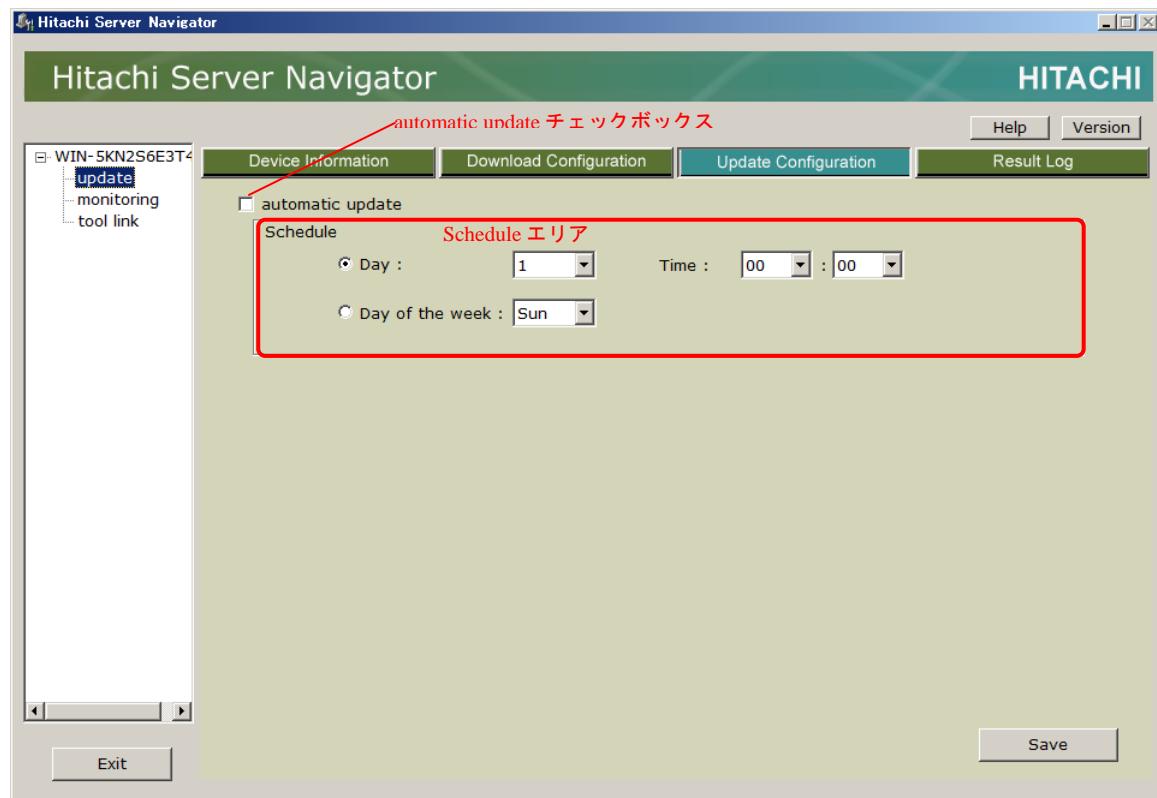


アップデート設定画面

全体メニュー ツリービューの「update」メニュー選択後、Update Configuration タブに対応している画面です。

自動アップデートの有効化と、自動アップデートのスケジュール設定が可能です。

アップデート設定画面について詳細を説明します。



automatic update チェックボックス

自動アップデートの可否を設定します。チェックが入っている状態では、スケジュールに従い自動アップデートを開始します。初期状態はチェックが入っていない状態です。



スケジュールに従い自動アップデートが行われた場合、automatic update チェックボックスのチェックは解除され、以後自動アップデートは行いません。再度自動実行が必要な場合は、自動アップデート設定を再設定してください。

Schedule エリア

以下の項目のアップデートスケジュールを表示および設定します。

項目	説明
Day ラジオボタン	アクティブにすると日付指定でのスケジュールが有効となります。 Day、Hour、Minutes ドロップダウンリストが設定可能となります。 Day of the week ラジオボタンと二者選一でアクティブに設定可能で、初期状態では Day ラジオボタンがアクティブです。
Day of the week ラジオボタン	アクティブにすると曜日指定でのスケジュールが有効となります。 Day of the week ドロップダウンリストが設定可能となります。
Day ドロップダウンリスト	日付を指定します。 未選択、01～31 の候補リストから設定可能です。 入力日が当日以下の場合、翌月の設定となります。31 日が存在しない月やうるう年を考慮して存在しない日付の場合、その翌月の 1 日の設定となります。
Day of the week ドロップダウンリスト	曜日を指定します。 未選択、Sun～Sat の候補リストから設定可能です。
Hour ドロップダウンリスト	時刻(時)を指定します。(24 時間指定) *
Minutes ドロップダウンリスト	時刻(分)を指定します。 *

* 時刻設定は、各アップデートスケジュールで共通です。



自動アップデートの開始時刻は、現在時刻から 1 時間以内に設定できません。

Save ボタン

アップデート設定を保存します。「ENTER」キーで代替可能です。

確認ダイアログが表示され、「はい」を選択することにより、アップデート設定が保存されます。「いいえ」を選択するとアップデート設定画面に戻ります。

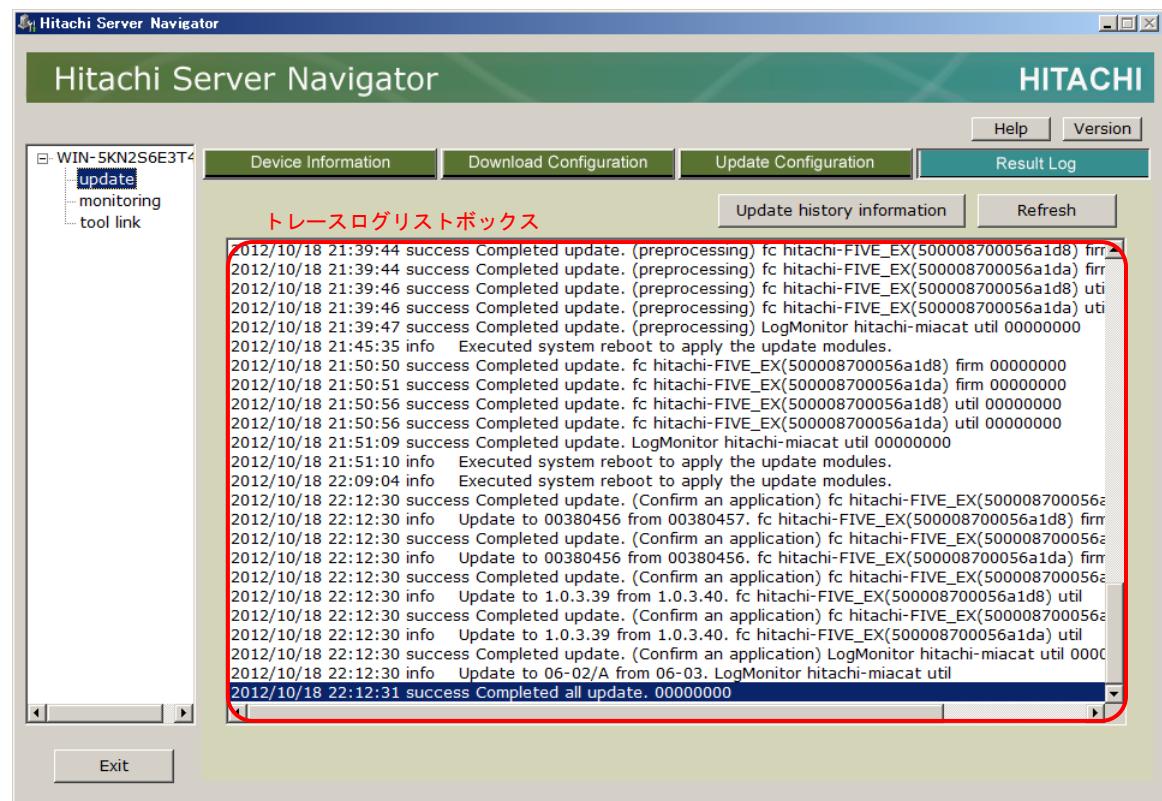


トレースログ画面

全体メニュー「ツリービュー」の「update」メニュー選択後、Result Log タブに対応している画面です。

Server Navigator のトレースログの参照が可能です。

トレースログ表示画面について詳細を説明します。



トレースログリストボックス

Server Navigator のトレースログを表示します。表示内容の詳細は、「Hitachi Server Navigator ユーザーズガイド アップデート機能」の「トレースログの表示内容の詳細」を参照してください。



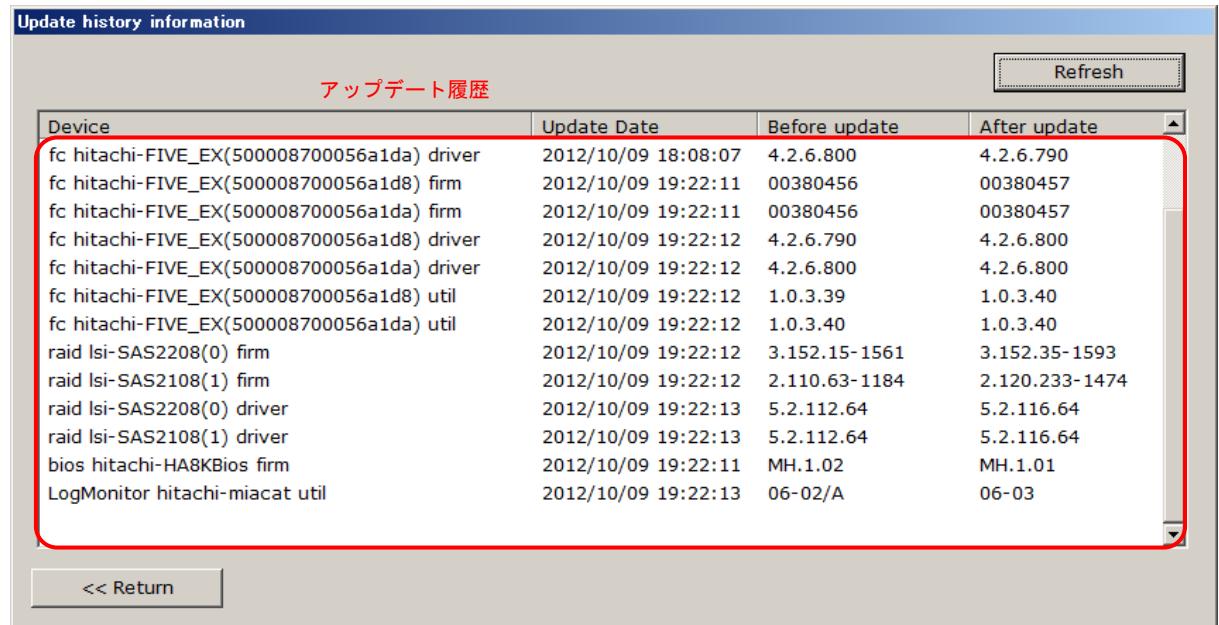
トレースログ画面の表示内容は自動更新しません。最新の情報を表示するには再表示を行ってください。

Refresh ボタン

トレースログ表示エリアの内容を再表示します。「ENTER」キーで代替可能です。

Update history information ボタン

アップデート履歴情報ダイアログ画面を表示します。



以下にアップデート履歴情報ダイアログ画面の表示項目を表記します。

項目	説明
Refresh ボタン	アップデートのログ情報を再表示します。
Device	プロバイダ種別、ベンダ名-ユニーク名(補足情報)、モジュール種別(firm/driver/util)を表示名を表示します。
Date	処理実行日時を表示します。
Before update	更新前のバージョンを表示を表示します。
After update	更新後のバージョンを表示を表示します。
Return ボタン	本画面を閉じて、前画面に戻ります。

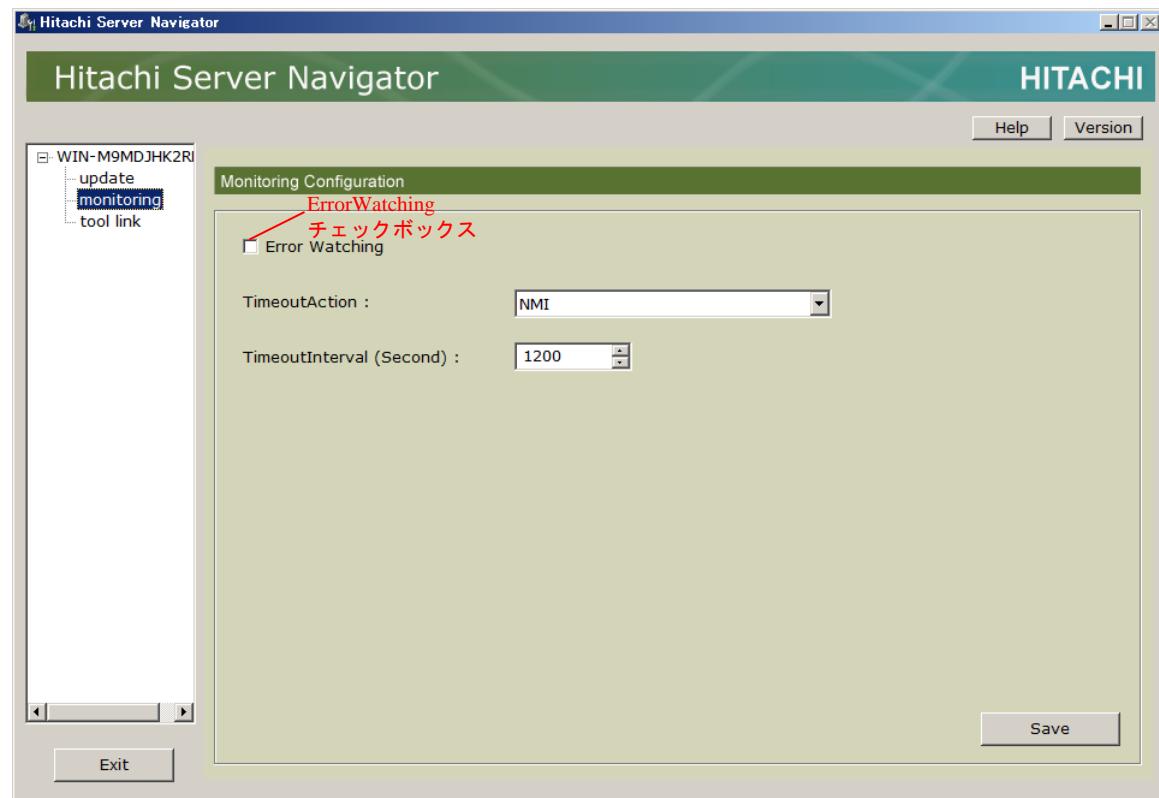
監視機能設定機能用GUI画面

全体メニュー「リービュー」の「monitoring」メニューに対応する、監視機能設定機能用の GUI 画面です。

Server Navigator の Alive Monitor 機能の設定が可能です。

監視機能設定画面

監視機能設定画面について詳細を説明します。



ErrorWatching チェックボックス

Server Navigator の Alive Monitor 機能を使用した、異常ログ監視の有無を設定します。チェックが入っている状態では、異常ログ監視を行います。初期状態はチェックが入っていない状態です。

TimeoutAction リストボックス

異常ログ監視でタイムアウト時に行うアクションをドロップダウンリストで選択します。

以下に TimeoutAction リストボックスで選択可能なアクションを表記します。初期状態は NMI です。

アクション	説明
NoAction	何も処理しません。
HardReset	サーバ装置の RESET ボタンを押したときと同じ処理をします。 なお、障害の状態によっては、正常にリセットできない場合があります。
PowerDown	サーバ装置の電源を OFF にします。
PowerCycle	サーバ装置の電源を OFF にして、すぐ ON にします。
NMI	NMI を発生させます。 この設定をする場合の注意事項については、「Hitachi Server Navigator ユーザーズガイド Alive Monitor 機能」を参照してください。

TimeoutInterval テキストボックス

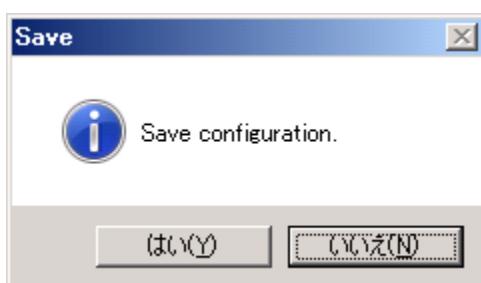
異常ログ監視のタイムアウトの間隔を設定します。

60~3600(秒)の値が設定可能です。初期状態は 1200(秒)です。

Save ボタン

アップデート設定を保存します。「ENTER」キーで代替可能です。

確認ダイアログが表示され、「はい」を選択することにより、アップデート設定が保存されます。「いいえ」を選択するとアップデート設定画面に戻ります。



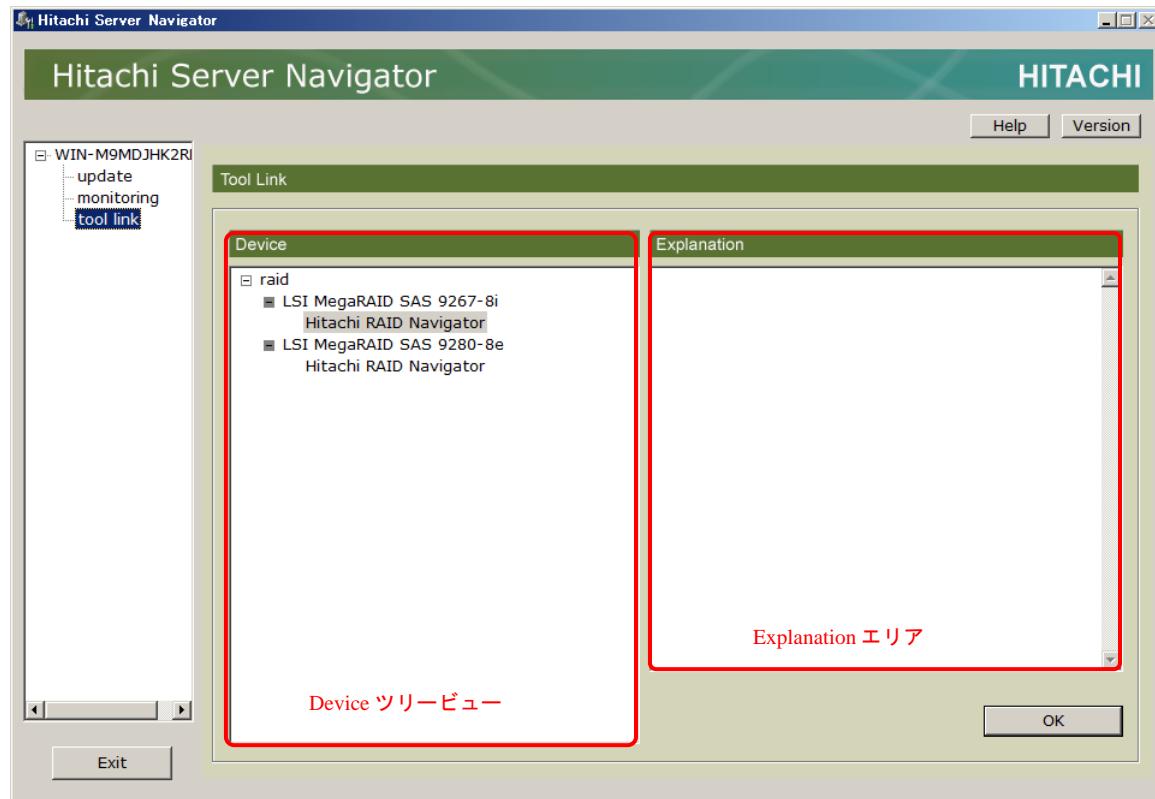
ツール連携機能用GUI画面

全体メニュー「ツリービュー」の「tool link」メニューに対応する、ツール連携機能用の GUI 画面機能です。

Server Navigator に連携した各種ツールの起動が可能です。

ツール連携画面

ツール連携画面について詳細を説明します。



Device ツリービュー

接続されているデバイスおよびツールをツリー表示します。Device ツリービューのツール名称をクリックすることにより、起動するツールが選択可能です。

Explanation エリア

Device ツリービューで選択したツールの説明が表示されるエリアです。

OK ボタン

Device ツリービューで選択したツールを起動します。

Server NavigatorのGUI操作の説明

この章では、Server Navigator GUI を使用した各種ユースケースにおける作業手順について説明します。

- [アップデート機能のGUI操作](#)
- [監視機能設定機能のGUI操作](#)
- [ツール連携機能のGUI操作](#)



BMC のアップデート行う場合は、JP/SC Agent または SEL Manager のインストールが必要です。インストールしていない場合は、何れかをインストールしてください。

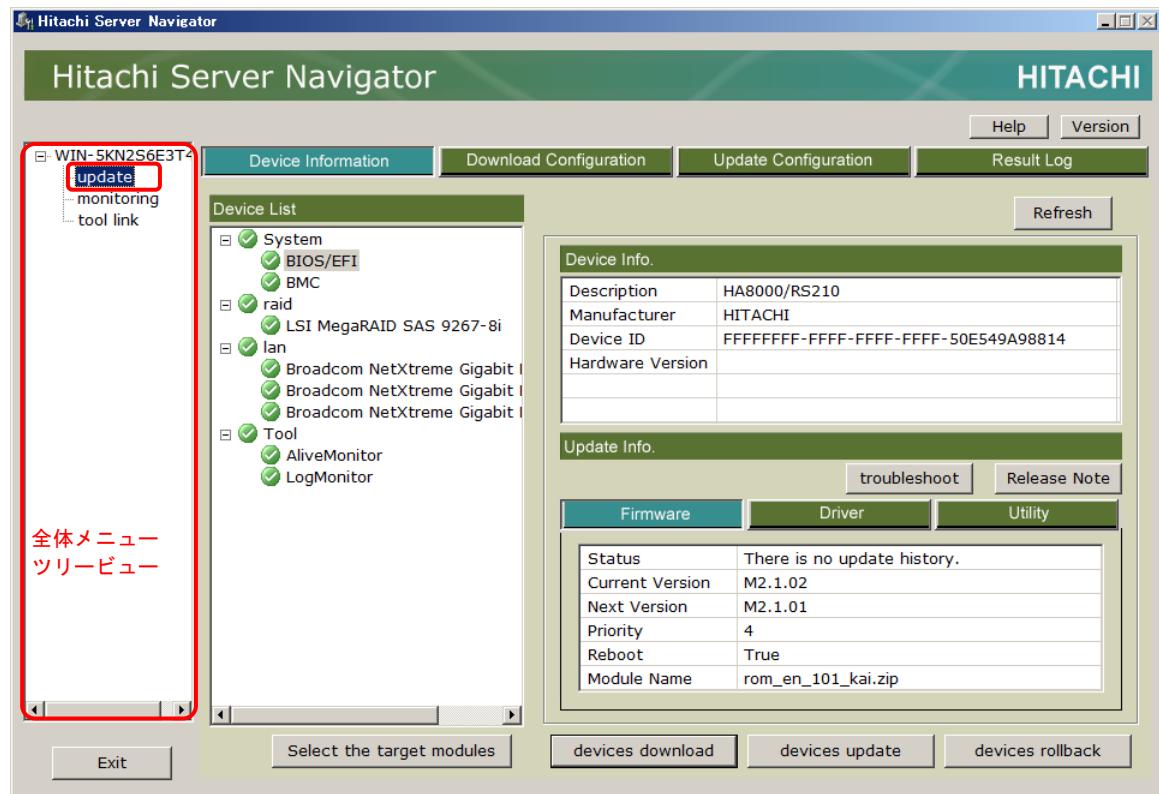


FC(Hitachi PCI Fibre Channel Adapter) をご使用のシステム構成で、本ユーティリティを使用する場合は、HFCTools のインストールが必要です。インストールしていない場合は、FC の状態表示およびアップデートを正常に行うことができません。

アップデート機能のGUI操作

GUIによるアップデート機能(ダウンロードやロールバック含む)の操作手順について説明します。

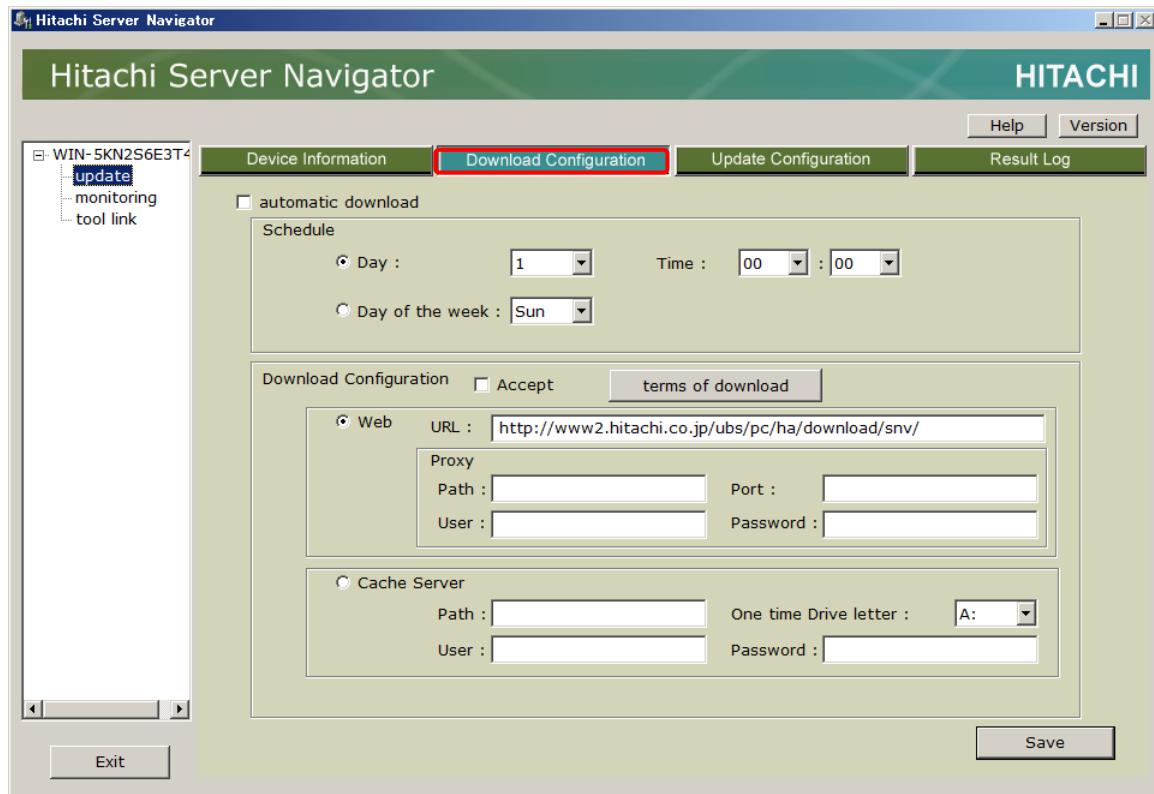
アップデート機能は、アップデート機能画面にて設定を行います。アップデート機能画面以外の画面を表示している場合は、全体メニュー ツリービューの「update」をクリックして、アップデート機能に切り替えてください。



自動ダウンロード設定

自動ダウンロードの設定手順について説明します。

自動ダウンロードは、ダウンロード設定画面にて設定を行います。ダウンロード設定画面以外の画面を表示している場合は、Download Configuration タブをクリックして、ダウンロード設定画面に切り替えてください。



自動ダウンロード機能を使用するには、本ユーティリティに対応している Web サイトと LAN 等で HTTP 接続可能である必要があります。

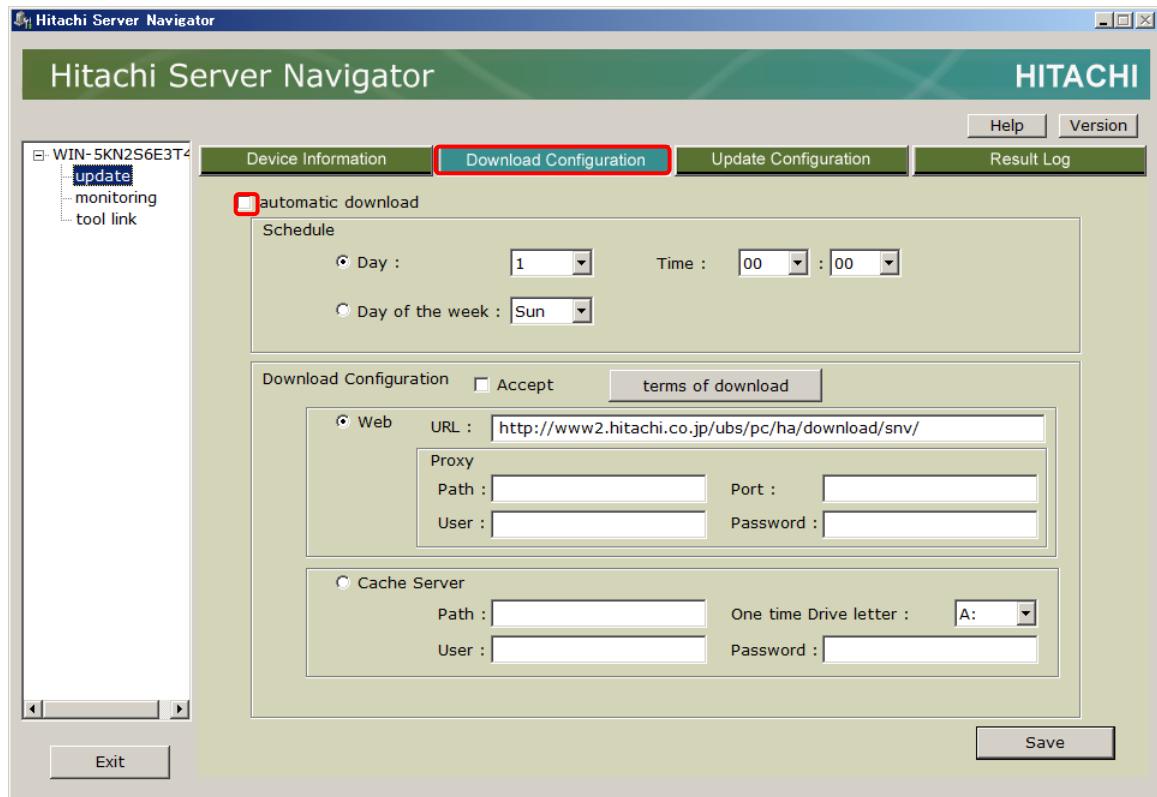


自動ダウンロード機能で設定する日時は、ダウンロード動作と他のアプリケーションとの競合を防ぐため、他のアプリケーションが動作していない状態を設定してください。また、管理ソフトなどによるシャットダウン時間から余裕を持った時間を設定してください。

自動ダウンロードの設定手順

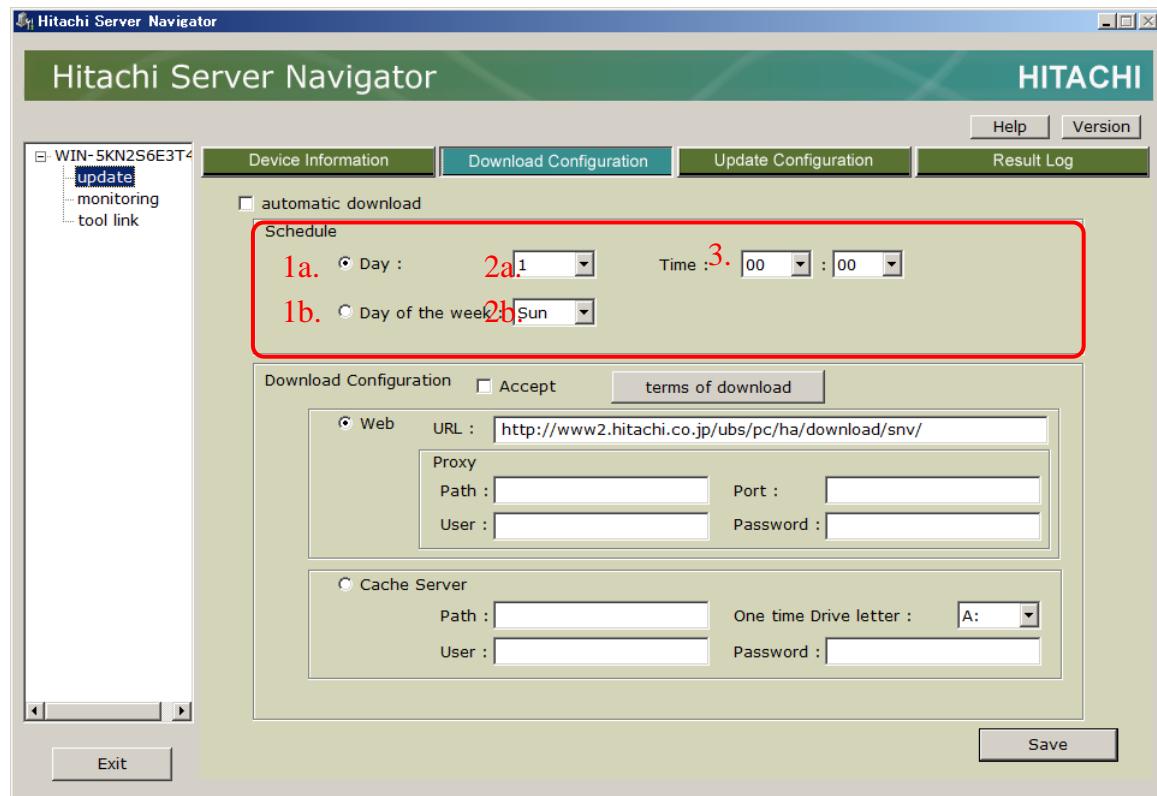
自動ダウンロードの有効化

自動ダウンロードを有効にするには、automatic download チェックボックスにチェックマークが付いている必要があります。チェックマークが付いていない場合は、チェックボックスをクリックしてチェックマークを付けてください。



スケジュールの設定

次に自動ダウンロードを行うスケジュールを設定します。スケジュールは、日付設定による一ヶ月以内実行、または、曜日設定による一週間以内実行が設定可能です。日付設定と曜日設定手順について説明します。



● 自動ダウンロードを一ヶ月以内の特定の日付に行いたい場合

- 1a. **Day** ラジオボタンをクリックしてください。
- 2a. Day ドロップダウンリストから日付を選択してください。
3. 開始時刻をドロップダウンリストから設定してください。

● 自動ダウンロードを一週間以内の特定の曜日に行いたい場合

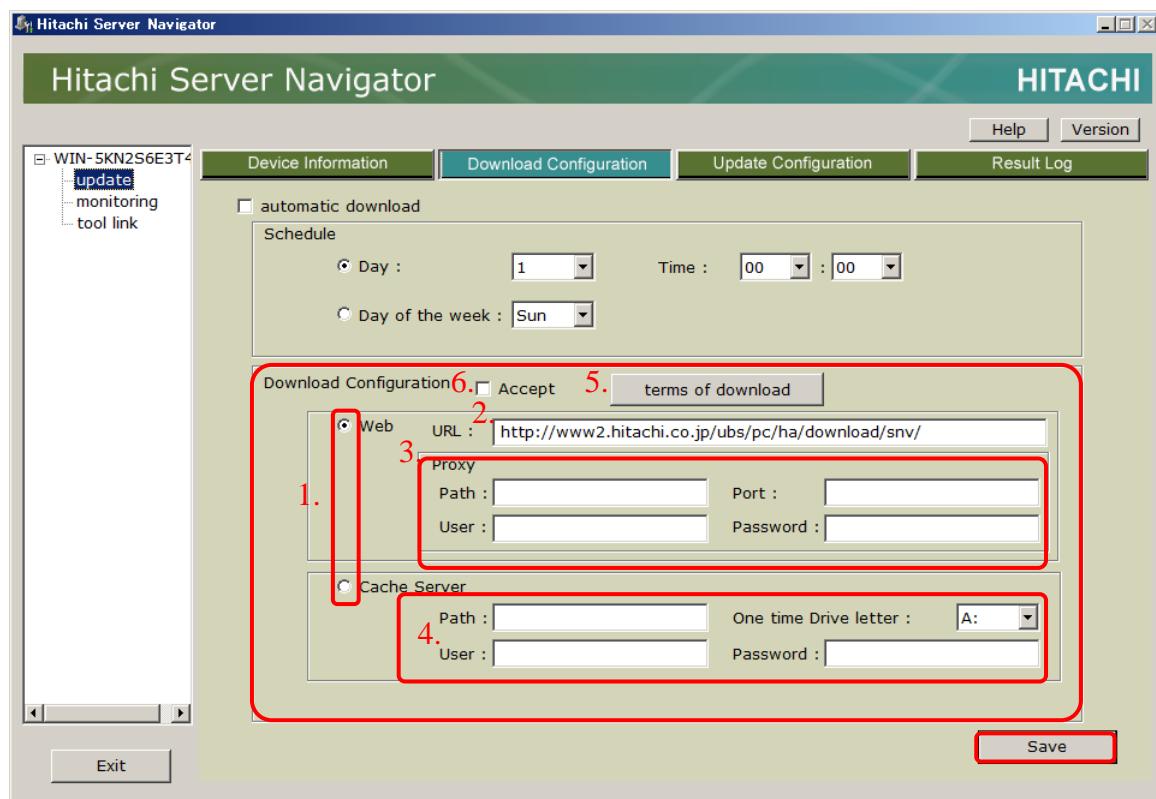
- 1b. **Day of the week** ラジオボタンをクリックしてください。
- 2b. Day of the week ドロップダウンリストから曜日を選択してください。
3. 開始時刻をドロップダウンリストから設定してください。



不慮のトラブル防止のため、自動ダウンロードの開始時刻は、現在時刻から 1 時間以内に設定することはできません。

ダウンロードの設定

次にダウンロードの設定を行います。次の手順で、ダウンロード接続先の設定、プロキシサーバの設定、使用許諾の確認、ダウンロード設定画面で行った設定の保存を行います。



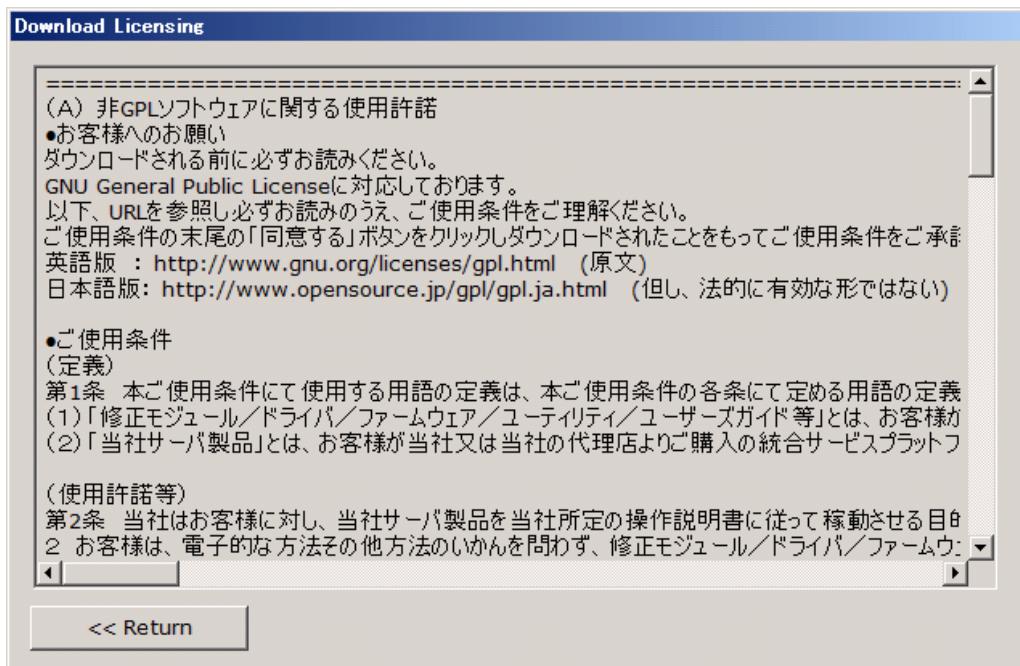
1. Web サイトからダウンロードを行う場合、Web ラジオボタンをクリックして 2.に進んでください。キャッシングサーバからダウンロードを行う場合は、Cache Server ラジオボタンをクリックして 4.に進んでください。
2. URL テキストボックスに本ユーティリティで接続する Web サイトの URL を入力してください。



接続する Web サイトは本ユーティリティに対応している必要があります。通常はデフォルトの設定のまま使用してください。

3. プロキシサーバを経由して Web サイトに接続する場合、Proxy エリアの Path、Port、User、Password テキストボックスに、プロキシサーバに接続するための URL、ポート番号、アカウント ID、パスワードを入力してください。プロキシサーバに接続しない場合は空欄にしてください。5.に進んでください。

4. Cache Server エリアの Path、One time Drive letter、User、Password テキストボックスに、キャッシングサーバに接続するためのキャッシングサーバのアドレス、一時使用ドライブレター、アカウント ID、パスワードを入力してください。
5. 「**terms of download**」ボタンをクリックしてください。使用許諾が表示されます。内容に問題なければ、「**Return**」ボタンを押してください。



6. 許諾に同意する場合のみ、Accept チェックボックスにチェックを入れてください。同意しない場合は、ダウンロード設定を保存することはできません。

ダウンロードの設定の保存

最後にダウンロード設定を保存します。「**Save**」ボタンをクリックすると確認ダイアログが表示されます。「はい」を選択することにより、ダウンロード設定が保存されます。「いいえ」を選択するとダウンロード設定画面に戻ります。



以上で、自動ダウンロード設定は完了です。

手動ダウンロード実行

ダウンロード指示機能を使用した、手動ダウンロード手順について説明します。

ダウンロード接続先として、ダウンロード設定画面で設定したダウンロード設定を使用する場合は、先に「自動ダウンロード設定」の「[ダウンロードの設定](#)」(P.4-6)を参照して、ダウンロードの設定を行ってください。



手動ダウンロード実行で Web サイトからダウンロードを行うには、本ユティリティに対応している Web サイトと LAN 等で HTTP 接続可能である必要があります。



手動ダウンロード実行で USB メモリ等のパスを指定してダウンロードを行うには、本ユティリティに対応している形式でアップデートモジュールが格納されている必要があります。詳細は、「Hitachi Server Navigator ユーザーズガイド アップデート機能」の「ダウンロードユティリティ」を参照してください。

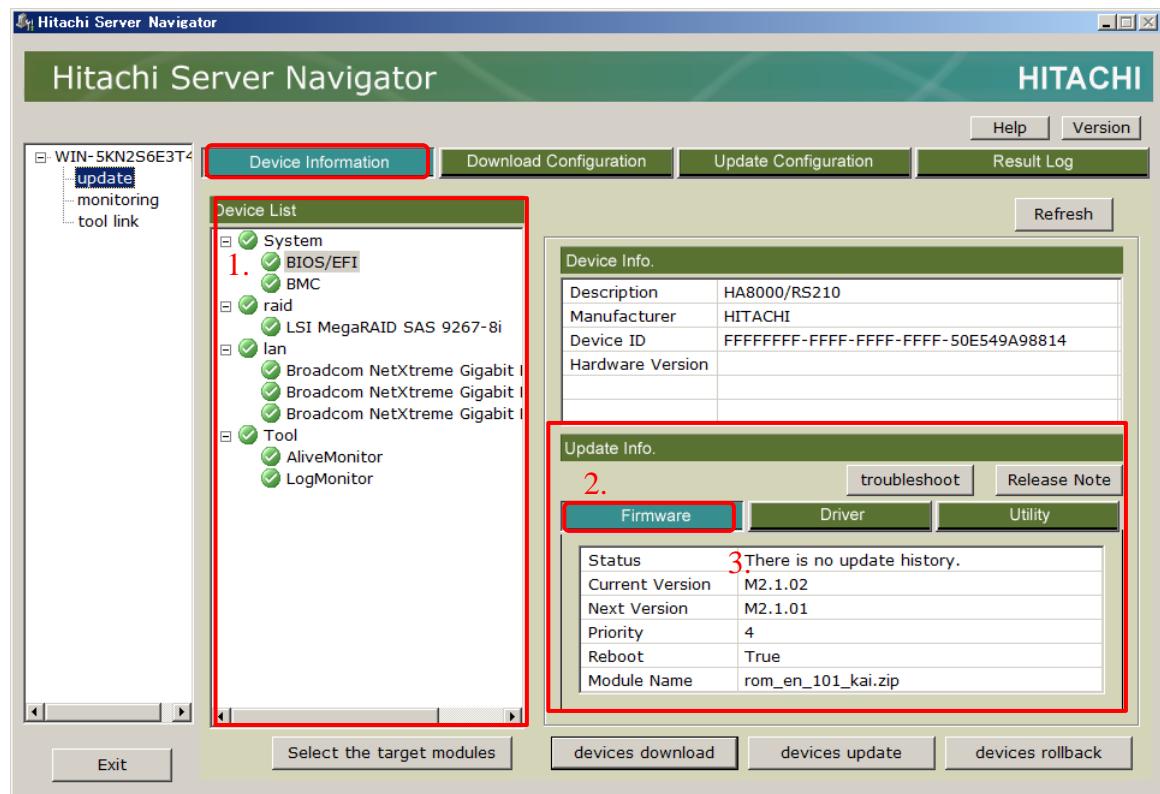


手動ダウンロード実行は、ダウンロード動作と他のアプリケーションとの競合を防ぐため、他のアプリケーションが動作していない状態で行ってください。

手動ダウンロードの実行手順

手動ダウンロード準備

Device Information タブをクリックして、デバイス情報画面に切り替えてください。



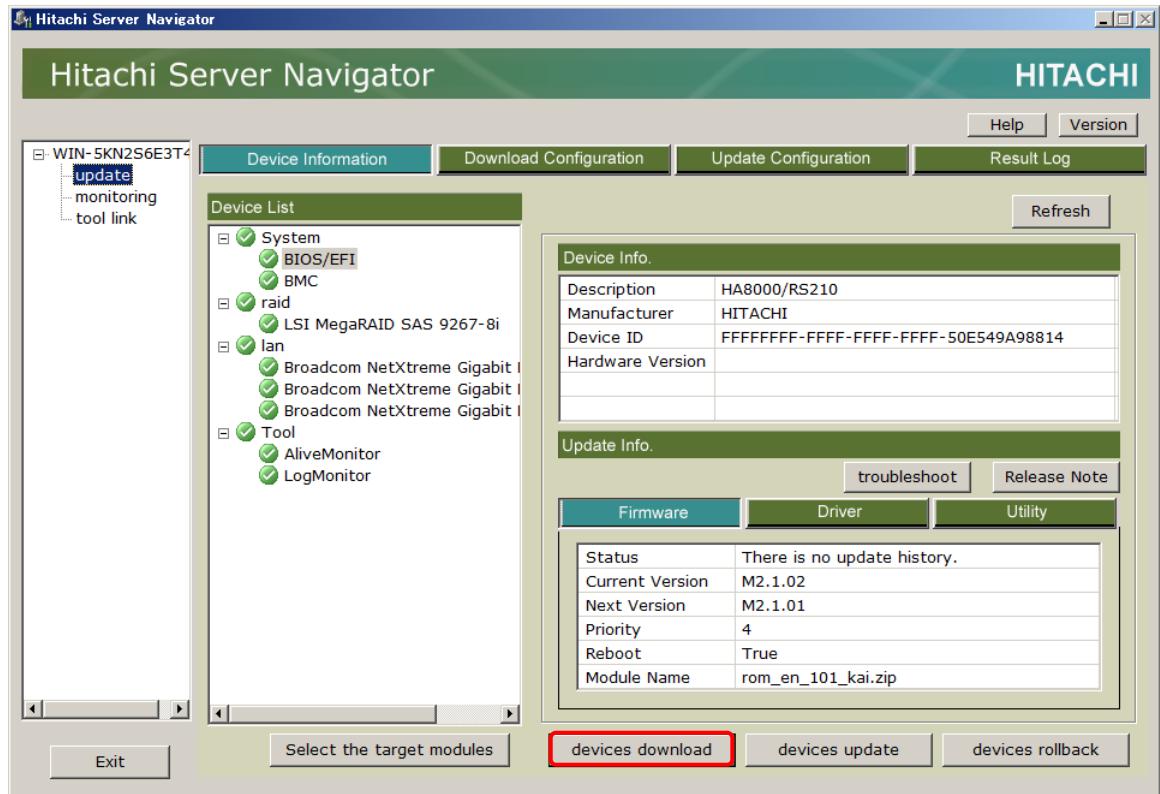
ダウンロードを行うデバイスのコンポーネントと、ツールに対し、下記 1. から 3. を行ってください。

1. Device List ツリービューから Update Info. エリアに表示するデバイスを選択してください。
2. Update Info. エリアに詳細情報を表示するコンポーネントに対応するタブをクリックしてください。
3. Status の表示が正常状態("There is no update history." または "Application success")であることを確認してください。

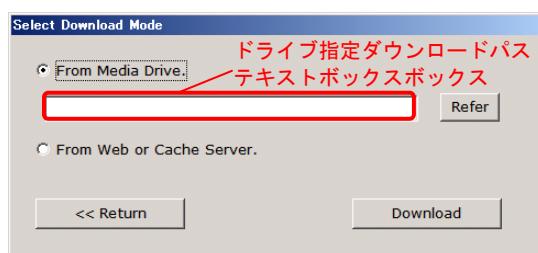


Statusの表示が正常状態以外の場合は、手動ダウンロードを含む、ダウンロード機能、アップデート機能、ロールバック機能は使用できません。「[トラブルシューティング](#)」(P.6-1)を参照して、状態の確認および必要に応じ回復を行ってください。

手動ダウンロード開始



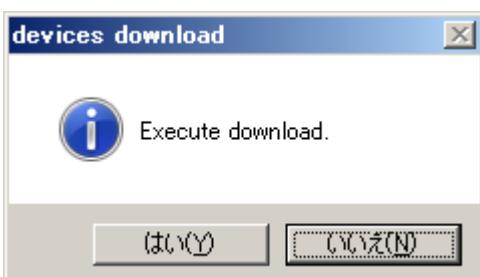
1. 「devices download」ボタンをクリックすると、Select Download Mode画面が開きます。ダウンロード設定を使用してWebサイトからダウンロードを行う場合は、「From web or Cache Server.」を選択して、3.に進んでください。



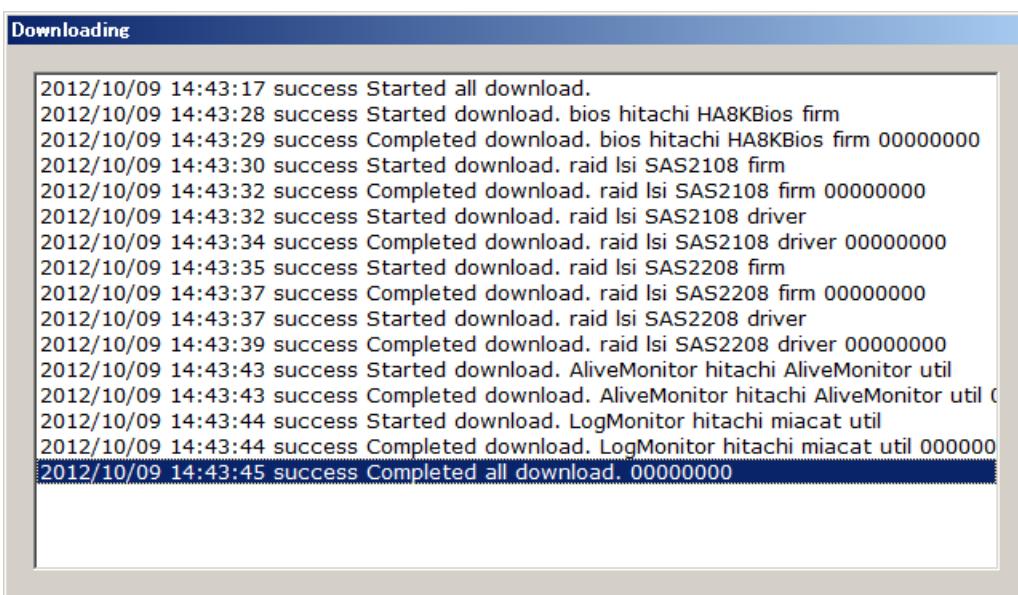
2. 「From Media Drive.」を選択した場合は、ドライブ指定ダウンロードパステキストボックスに設定されたパスからダウンロードを行います。ドライブ指定ダウンロードパステキストボックスに直接パスを入力するか、Referボタンを押してフォルダ参照ダイアログからパスを選択します。



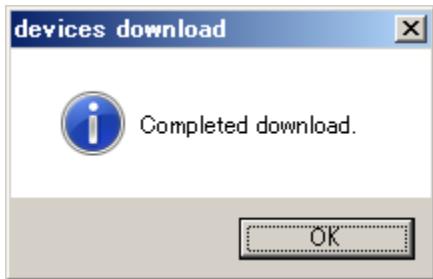
3. **Download** ボタンをクリックすると、誤動作防止のための確認ダイアログが表示されます。



4. 「はい」を選択するとターゲットモジュール選択機能で設定されたモジュールのダウンロード処理を開始します。「いいえ」を選択すると Select Download Mode 画面に戻ります。
5. 手動ダウンロード実行中は、ダウンロード実行中画面を表示し、ダウンロード状況を画面上に表示します。ダウンロード実行中画面を表示中は全ての操作を行ふことができません。



- 手動ダウンロードが完了した場合、ダウンロード実行中画面は消え、ダウンロード完了ダイアログが表示されます。「OK」を選択するとデバイス情報画面に戻ります。



以上で、手動ダウンロードは完了です。

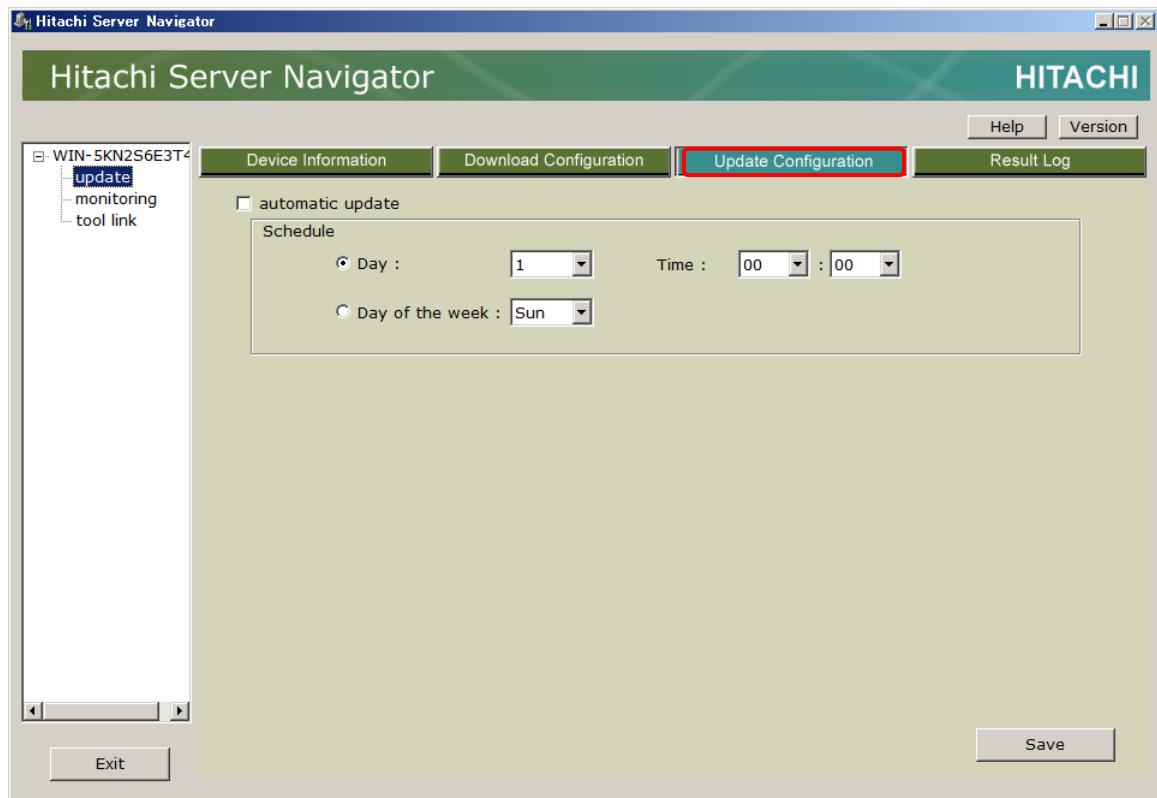


ダウンロード先と接続失敗しない限り、実際にモジュールのダウンロードが行われたかに依存せず、"Completed download." のダイアログが表示されます。接続失敗時のメッセージについては、「Hitachi Server Navigator ユーザーズガイド アップデート機能」の「ダイアログメッセージ一覧」を参照してください。

自動アップデート設定

自動アップデートの設定手順について説明します。

自動アップデートは、アップデート設定画面にて設定を行います。アップデート設定画面以外の画面を表示している場合は、Update Configuration タブをクリックして、アップデート設定画面に切り替えてください。



アップデート実行時は必要に応じシステムリブートを行うため、自動アップデート機能で設定する日時は、他のアプリケーションが動作していない状態を設定してください。また、管理ソフトなどによるシャットダウン時間から余裕を持った時間を設定してください。



アップデート実行時のシステムリブートの有無は、自動ダウンロードまたは手動ダウンロードでダウンロードが実行済みの場合、「[デバイス情報画面](#)」(P.3-7)で確認可能です。また、「[システムリブート回数の目安](#)」(P.1-9)で確認可能です。



BIOS/EFI、Log Monitor、Alive Monitor のダウングレードは未サポートのため、ダウングレードとなるアップデートは行いません。エラーとはなりませんが処理をスキップします。



BMC のアップデート中は、BMC フームウェアのバージョンによって、BMC の Web コンソールが正常に動作しない場合があります。また、Web コンソール以外の BMC の機能についても、処理速度が遅くなることがあります。

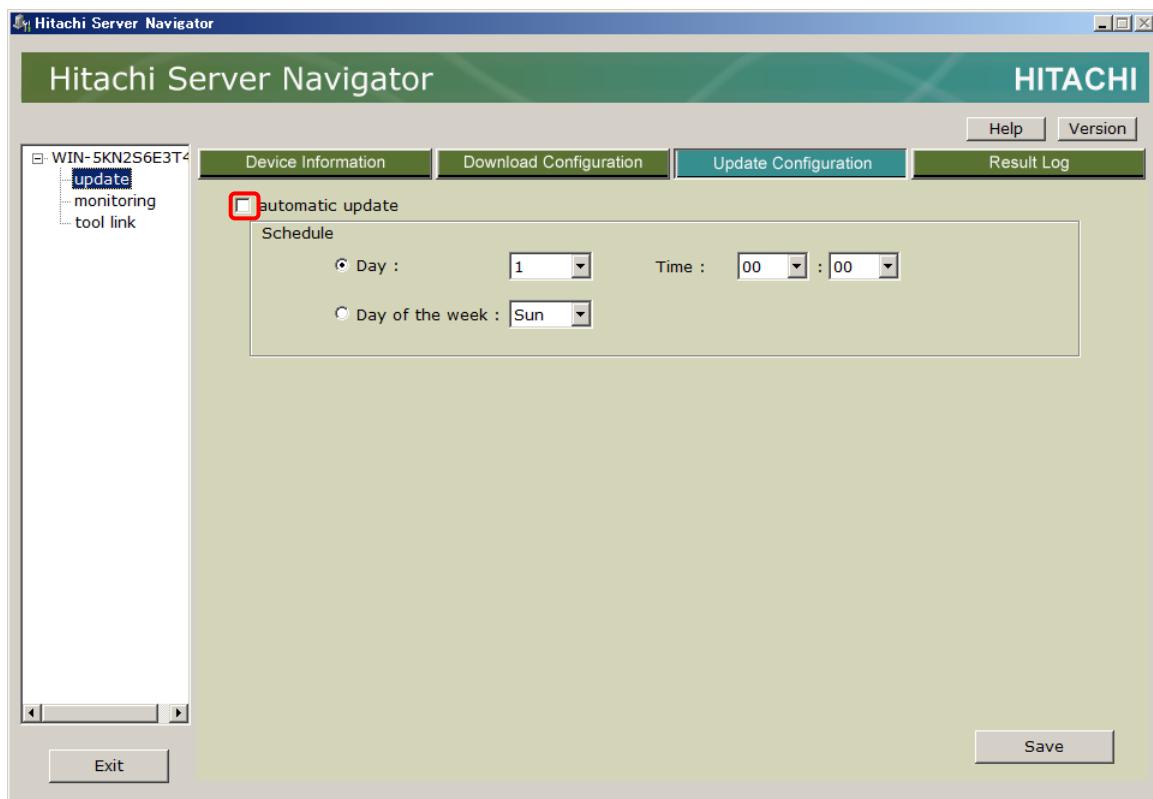
BMC フームウェアのアップデート中には、できるだけ BMC の機能の使用は控え、BMC フームウェアのアップデートが完了後、システムを再起動した後で、ご使用いただくようにお願いします。

万一、Web コンソールからの設定操作を実施中などに、BMC フームウェアの更新が実行された場合には、BMC フームウェアのアップデート完了後に、設定内容をご確認いただくようにお願い致します。

自動アップデートの設定手順

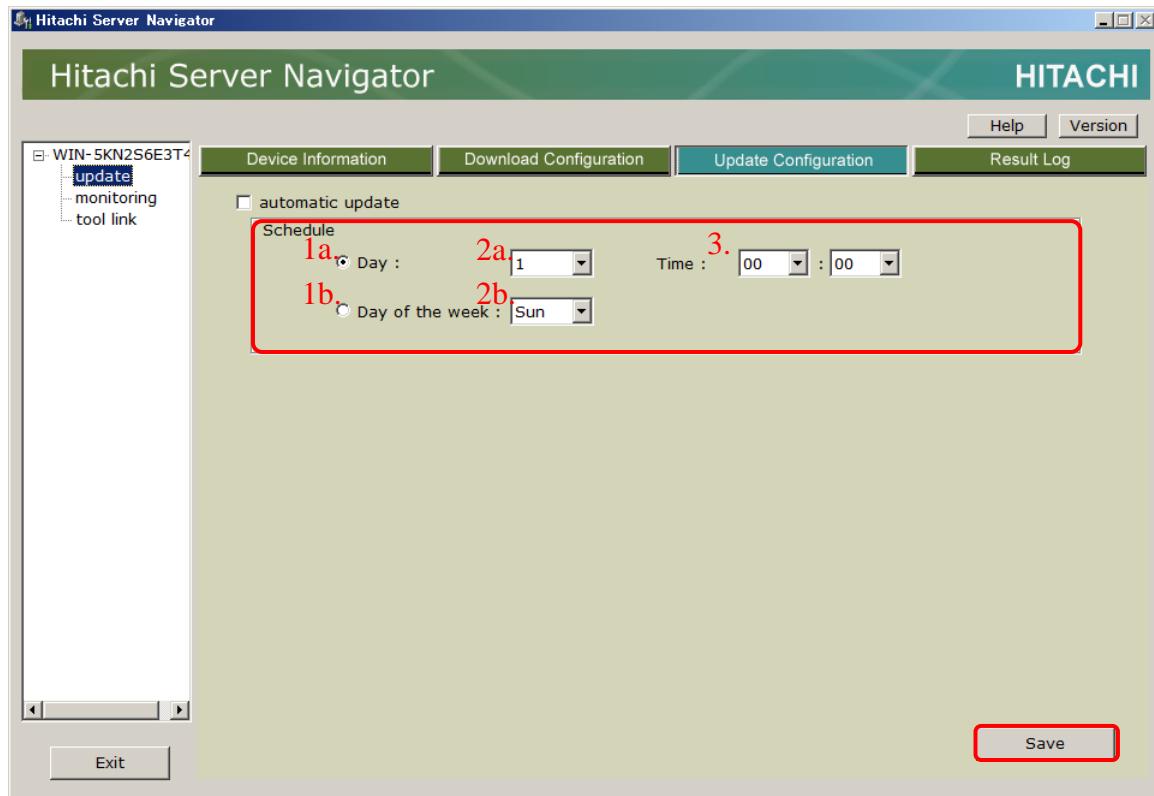
自動アップデートの有効化

自動アップデートを有効にするには、automatic update チェックボックスにチェックマークが付いている必要があります。チェックマークが付いていない場合は、チェックボックスをクリックしてチェックマークを付けてください。



スケジュールの設定

次に自動アップデートを行うスケジュールを設定します。日付設定または曜日設定の設定手順について説明します。



● 自動アップデートを一ヶ月以内の特定の日付に行いたい場合

- 1a. **Day** ラジオボタンをクリックしてください。
- 2a. Day ドロップダウンリストから日付を選択してください。
3. 開始時刻をドロップダウンリストから設定してください。

● 自動アップデートを一週間以内の特定の曜日に行いたい場合

- 1b. **Day of the week** ラジオボタンをクリックしてください。
- 2b. Day of the week ドロップダウンリストから曜日を選択してください。
3. 開始時刻をドロップダウンリストから設定してください。



不慮のトラブル防止のため、開始時刻は現在時刻から 1 時間以内に設定することはできません。

アップデートの設定の保存

最後にアップデート設定を保存します。Save ボタンをクリックすると確認ダイアログが表示されます。「はい」を選択することにより、アップデート設定が保存されます。「いいえ」を選択するとアップデート設定画面に戻ります。



以上で、自動アップデート設定は完了です。



自動アップデートの実行状況の確認は、OS のイベントログで確認可能です。“Started automatic update.” が開始を示し、“Completed automatic update.” が完了を示します。イベントログの詳細は「Hitachi Server Navigator ユーザーズガイド アップデート機能」の「イベントログ一覧」を参照してください。



自動アップデート実行中は、Server Navigator GUI を起動しないでください。アップデート処理が異常終了する場合があります。



自動アップデートが正常に行われたかの確認は、トレースログ画面で確認可能です。トレースログの内容については、「Hitachi Server Navigator ユーザーズガイド アップデート機能」の「トレースログの表示内容の詳細」を参照してください。

手動アップデート実行

アップデート指示機能を使用した、手動アップデート手順について説明します。



アップデート実行時は必要に応じシステムリブートを行うため、手動アップデート実行は、他のアプリケーションが動作していない状態で行ってください。



アップデート実行時のシステムリブートの有無は、自動ダウンロードまたは手動ダウンロードでダウンロードが実行済みの場合、「[デバイス情報画面](#)」(P.3-7)で確認可能です。また、「[システムリブート回数の目安](#)」(P.1-9)で確認可能です。



アップデート中はアップデート実行中画面を表示し操作不可となりますが、プロセス強制終了等で中断しないでください。中断するとシステムが正常に起動しなくなる可能性があります。



BIOS/EFI、Log Monitor、Alive Monitor のダウングレードは未サポートのため、ダウングレードとなるアップデートは行いません。エラーとはなりませんが処理をスキップします。



BMC のアップデート中は、BMC フームウェアのバージョンによって、BMC の Web コンソールが正常に動作しない場合があります。また、Web コンソール以外の BMC の機能についても、処理速度が遅くなることがあります。

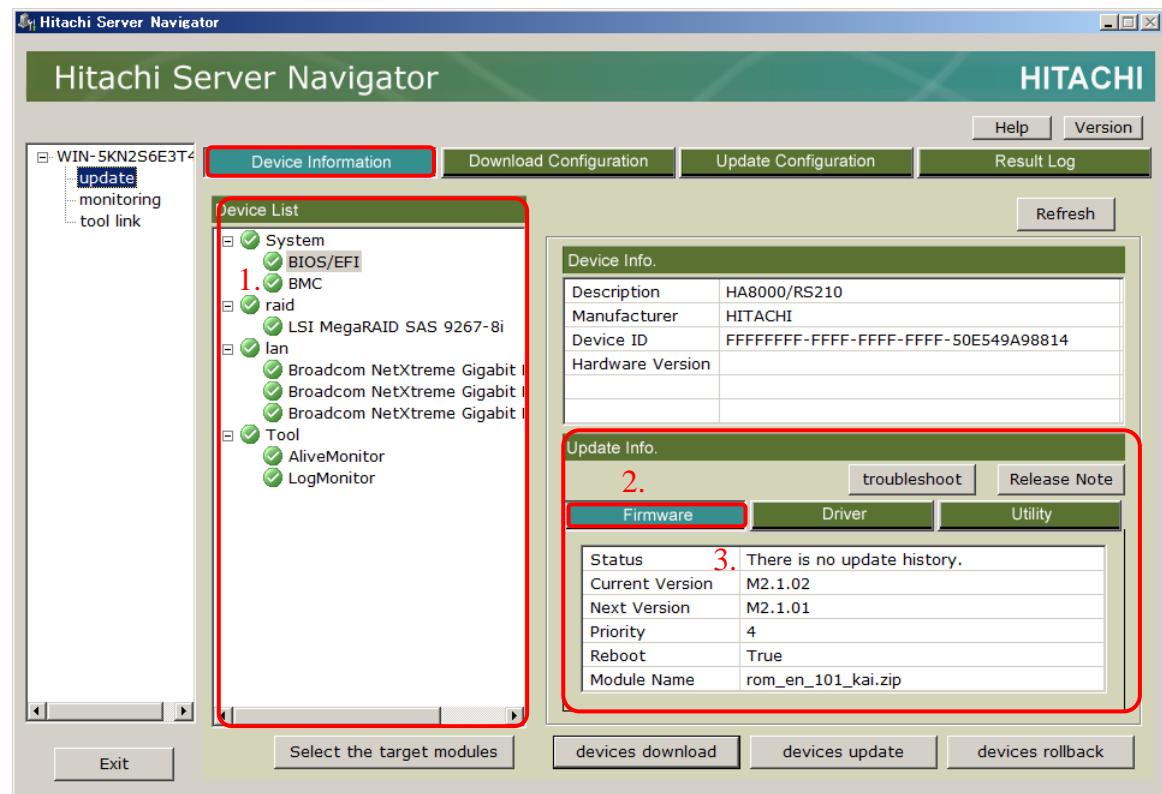
BMC フームウェアのアップデート中には、できるだけ BMC の機能の使用は控え、BMC フームウェアのアップデートが完了後、システムを再起動した後で、ご使用いただくようにお願いします。

万一、Web コンソールからの設定操作を実施中などに、BMC フームウェアの更新が実行された場合には、BMC フームウェアのアップデート完了後に、設定内容をご確認いただくようにお願い致します。

手動アップデートの実行手順

手動アップデートの準備

Device Information タブをクリックして、デバイス情報画面に切り替えてください。



アップデートを行うデバイスのコンポーネントと、ツールに対し 1.から 3.を行ってください。

1. Device List ツリービューから Update Info. エリアに表示するデバイスを選択してください。
2. Update Info. エリアに詳細情報を表示するコンポーネントに対応するタブをクリックしてください。
3. Status の表示が正常状態("There is no update history." または "Application success")であることを確認してください。

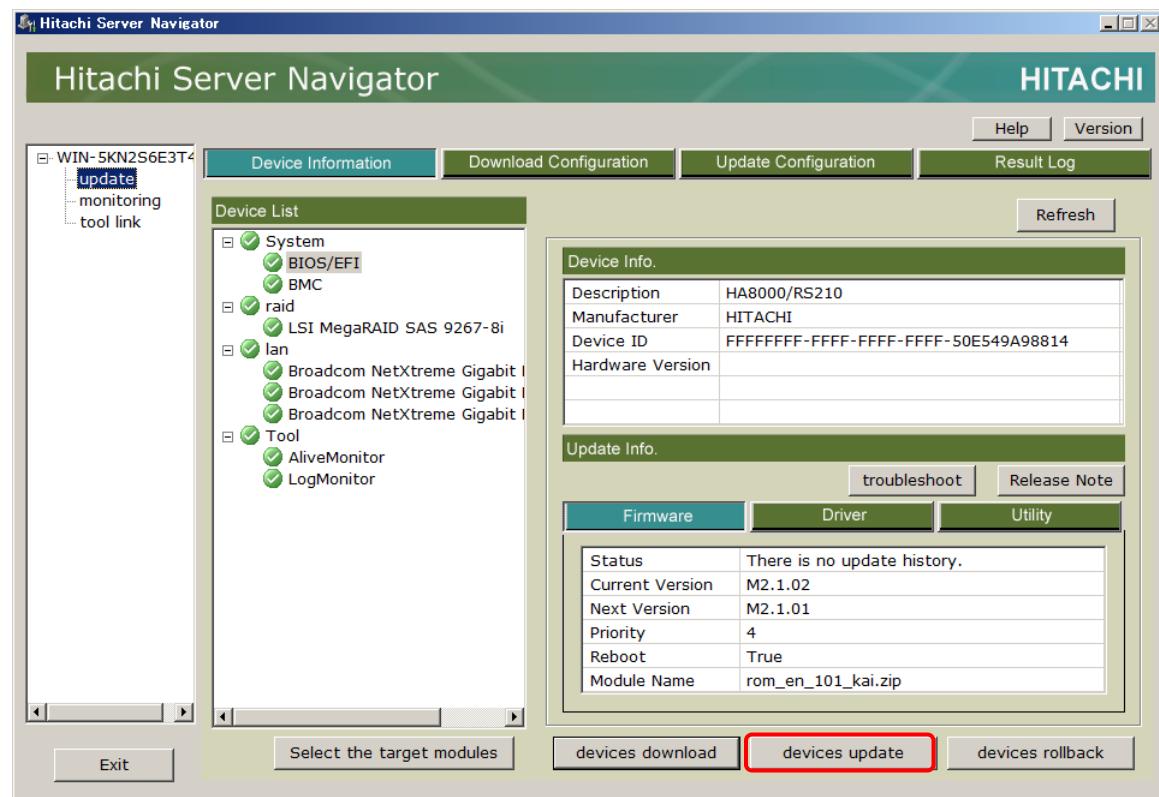


Statusの表示が正常状態以外の場合は、手動アップデートを含む、ダウンロード機能、アップデート機能、ロールバック機能は使用できません。「[トラブルシューティング](#)」(P.6-1)を参照して、状態の確認および必要に応じ回復を行ってください。

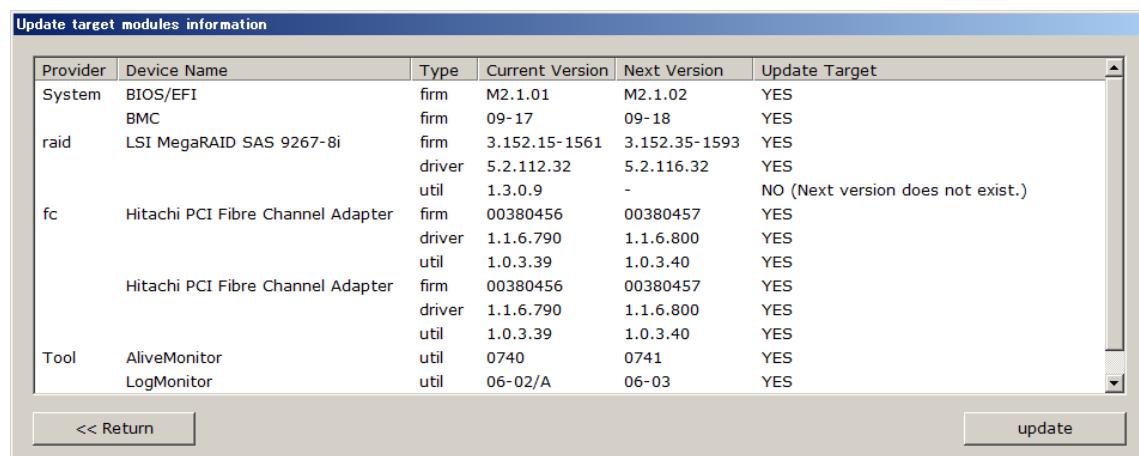


各コンポーネントの Reboot の値を確認することにより、アップデート処理中に OS のリブートが発生するかどうかが確認できます。1つでも True の値があった場合、リブートが発生します。

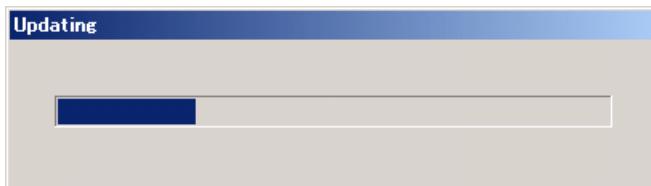
手動アップデート開始



1. 「devices Update」ボタンをクリックすると、Update target modules information 画面が開きますので、アップデートを実行するか、中止するかを選択します。



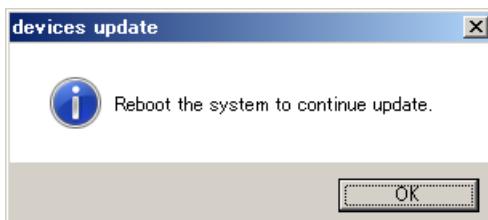
2. 「update」を選択するとターゲットモジュール選択機能で設定され、アップデート条件を満たしているモジュールの手動アップデート処理を開始します。「Return」を選択するとデバイス情報画面に戻ります。
3. 手動アップデート実行中は、アップデート実行中画面を表示し、アップデート状況を画面上に表示します。アップデート実行中画面を表示中は全ての操作を行うことができません。



アップデート処理の実行中は、一時的に応答が遅くなる場合がありますが問題ございません。



アップデート実行後、リブートが必須なアップデートモジュールを含む場合、システムは最大4回、自動でリブートを行います。ただし、1回目は下記画面を表示します。OKを選択することによりシステムリブートを開始します。



以後、アップデートが完了するまで操作を行わないでください。BMCのアップデートを含む場合、数十秒間のサーバ電源OFF期間がありますが、自動で再起動されますので、電源ボタンによる電源ONなど行なわないようお願いします。

手動アップデート実行手順はこれで終了となります。アップデート結果はGUIを起動後、トレースログ画面で確認してください。

4. 手動アップデートが正常終了した場合、アップデート実行中画面は消え、アップデート完了ダイアログが表示されます。OKを選択するとデバイス情報画面に戻ります。



以上で、手動アップデートは完了です。



手動アップデートの実行状況の確認は、OS のイベントログで確認可能です。“Started all update.” が開始を示し、“Completed all update.” が完了を示します。イベントログの詳細は「Hitachi Server Navigator ユーザーズガイド アップデート機能」の「イベントログ一覧」を参照してください。



手動アップデート実行中は、Server Navigator GUI を起動しないでください。アップデート処理が異常終了する場合があります。



手動アップデートが正常に行われたかの確認は、トレースログ画面で確認可能です。トレースログの内容については、「Hitachi Server Navigator ユーザーズガイド アップデート機能」の「トレースログの表示内容の詳細」を参照してください。

手動ロールバック実行

手動ロールバックを行うと一つ前のアップデート状態に戻ります。ロールバック指示機能を使用した、手動ロールバック手順について説明します。



ロールバックはアップデート後、システムが不安定となった場合の回復手段としての使用を想定しています。必要ない場合は行わないでください。



ロールバックはアップデート前の稼動バージョンのアップデートモジュールを使用して行います。そのため、ロールバック前のアップデートをする前に、アップデート前の稼動バージョンのアップデートモジュールをダウンロードしておく必要があります。



手動ロールバック実行では、直前にアップデートしたコンポーネントのみではなく、Select the target modules 画面で選択されている全てのコンポーネントに対して、保存されている過去バージョンに戻します。ロールバックを行いたいデバイスまたはツールのみ選択してから、ロールバックを行ってください。



手動ロールバック実行前のアップデート状態に戻すには、再び手動アップデート実行してください。



ロールバック実行時は必要に応じシステムリブートを行うため、手動ロールバック実行は、他のアプリケーションが動作していない状態で行ってください。RAID をロールバックする場合は I/O の少ない状態で行ってください。



ロールバック実行時のシステムリブートの有無は、「[デバイス情報画面](#)」(P.3-7)で確認可能です。また、「[システムリブート回数の目安](#)」(P.1-9)で確認可能です。



リブート有りの場合で、システムの運用上システムリブートができない状態の場合は、手動ロールバック実行を行わないでください。



BIOS/EFI、Log Monitor、Alive Monitor のダウングレードは未サポートのため、ダウングレードとなるロールバックは行いません。エラーとはなりませんが処理をスキップします。



BMC のアップデート中は、BMC フームウェアのバージョンによって、BMC の Web コンソールが正常に動作しない場合があります。また、Web コンソール以外の BMC の機能についても、処理速度が遅くなることがあります。

BMC フームウェアのアップデート中には、できるだけ BMC の機能の使用は控え、BMC フームウェアのアップデートが完了後、システムを再起動した後で、ご使用いただくようにお願いします。

万一、Web コンソールからの設定操作を実施中などに、BMC フームウェアの更新が実行された場合には、BMC フームウェアのアップデート完了後に、設定内容をご確認いただくようにお願い致します。

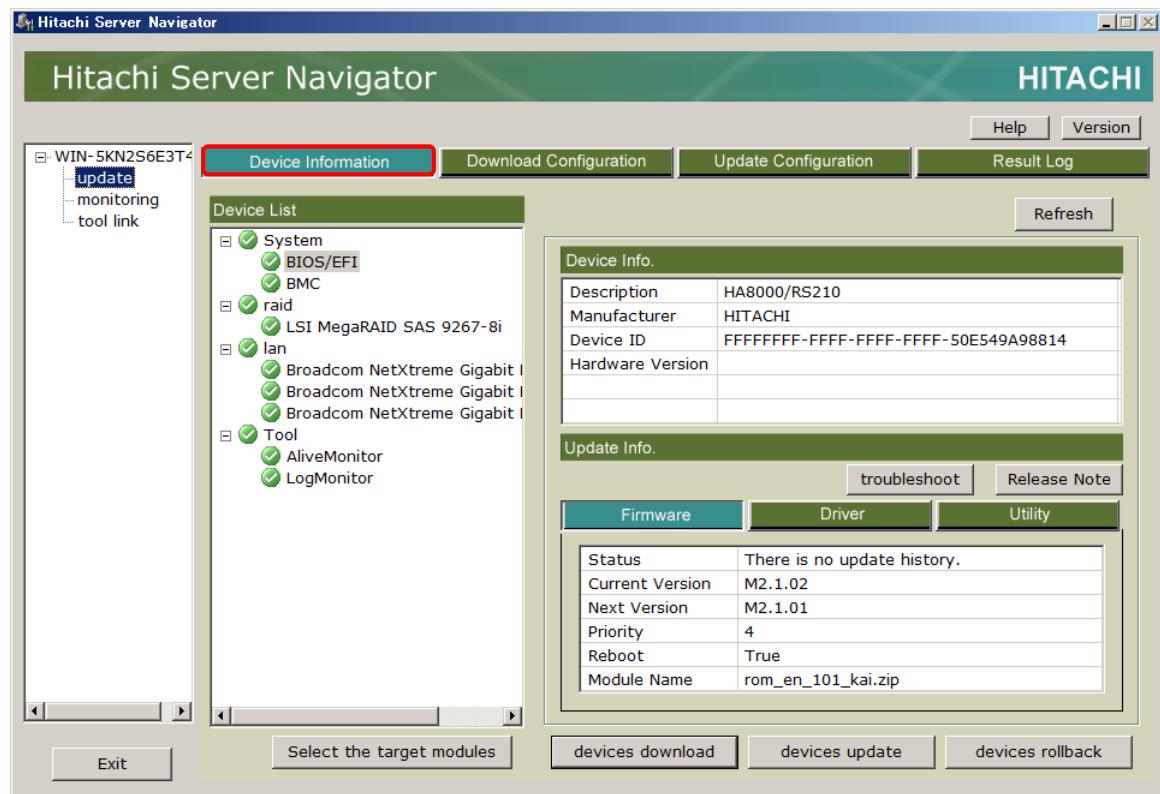


ロールバックは Windows の Safe Mode(セーフモード)では実行できません。通常起動してから実行してください。

手動ロールバックの実行手順

手動ロールバックの準備

Device Information タブをクリックして、デバイス情報画面に切り替えてください。



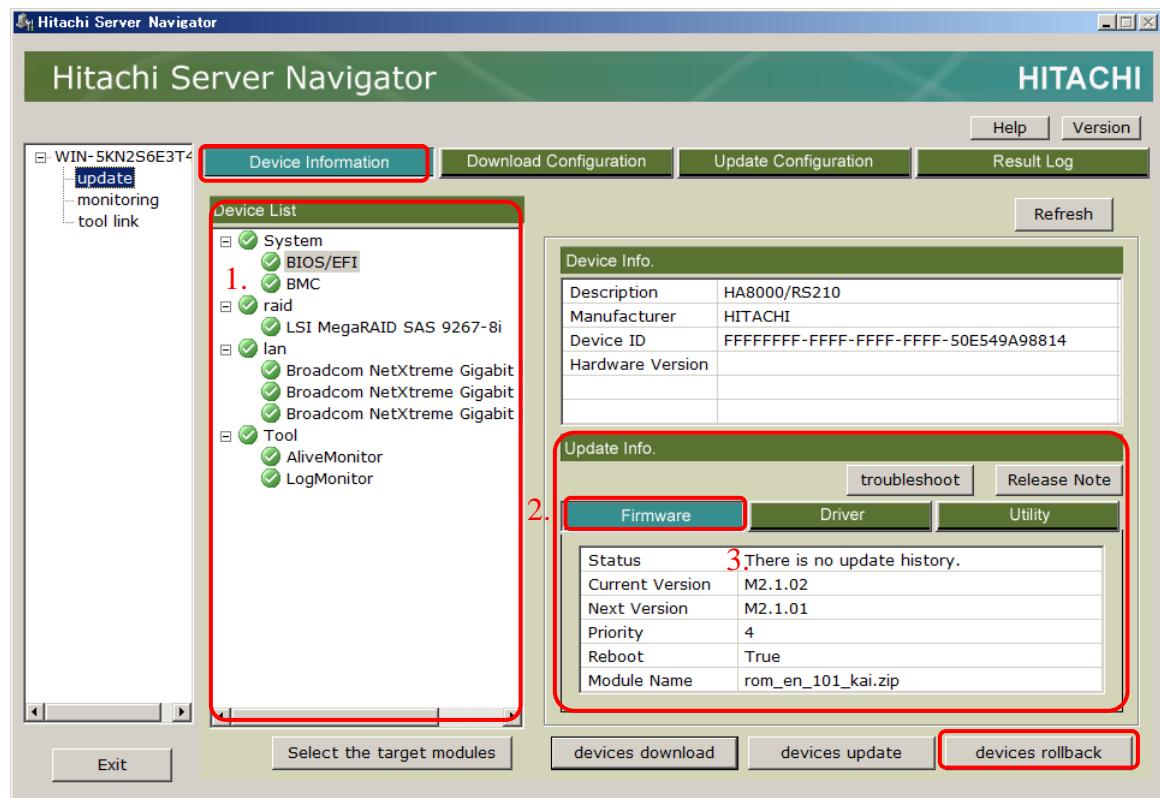
すべてのデバイスとコンポーネントに対し 1. から 3. を行ってください。

1. Device List ツリービューから Update Info. エリアに表示するデバイスを選択してください。
2. Update Info. エリアに詳細情報を表示するコンポーネントに対応するタブをクリックしてください。
3. Status の表示が正常状態("There is no update history." または "Application success")であることを確認してください。



Statusの表示が正常状態以外の場合は、手動ロールバックを含む、ダウンロード機能、アップデート機能、ロールバック機能は使用できません。「[トラブルシューティング](#)」(P.6-1)を参照して、状態の確認および必要に応じ回復を行ってください。

手動ロールバックの開始

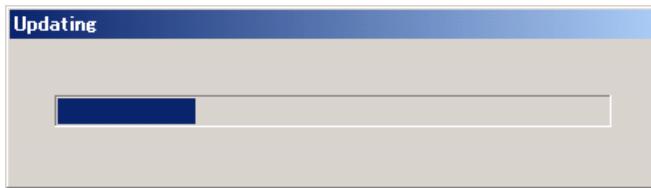


1. 「devices rollback」ボタンをクリックすると、Update target modules information 画面が開きますので、アップデートを実行するか、中止するかを選択します。

Update target modules information							
Provider	Device Name	Type	Current Version	Next Version	Update Target		
raid	LSI MegaRAID SAS 9267-8i	firm	3.152.35-1593	3.152.15-1561	YES		
		driver	5.2.116.64	5.2.112.64	YES		
		util	1.2.0.18	-	NO (Next version does not exist.)		
lan	Broadcom NetXtreme Gigabit Eth...	firm	2.120.233-1...	2.110.63-1184	YES		
		driver	5.2.116.64	5.2.112.64	YES		
		util	1.2.0.18	-	NO (Next version does not exist.)		
fc	Broadcom NetXtreme Gigabit Eth...	firm	5718-v1.43	-	NO (Next version does not exist.)		
		driver	15.0.0.21	-	NO (Next version does not exist.)		
		util	v15.2.20.0	-	NO (Next version does not exist.)		
	Hitachi PCI Fibre Channel Adapter	firm	5718-v1.43	-	NO (Next version does not exist.)		
		driver	15.0.0.21	-	NO (Next version does not exist.)		
		util	v15.2.20.0	-	NO (Next version does not exist.)		
						rollback	
<< Return							

2. 「rollback」を選択するとターゲットモジュール選択機能で設定され、アップデート条件を満たしているモジュールの手動ロールバック処理を開始します。「return」を選択するとデバイス情報画面に戻ります。

3. 手動ロールバック実行中は、アップデート実行中画面を表示し、ロールバック状況を画面上に表示します。アップデート実行中画面を表示中は全ての操作を行なうことができません。



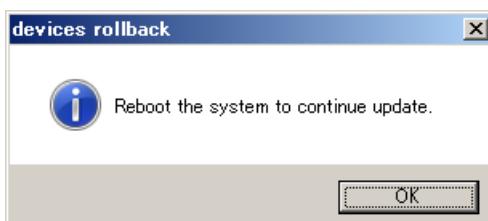
ロールバックでもデバイスに対する処理は共通のため、アップデート実行中画面が表示されます。



ロールバック処理の実行中は、一時的に応答が遅くなる場合がありますが問題ございません。



ロールバック後、リブートが必須なアップデートモジュールを含む場合、システムは最大3回、自動でリブートを行います。ただし、1回目は下記画面を表示しますので、OKを選択してください。



以後、ロールバックが完了するまで操作を行わないでください。BMC のアップデートを含む場合、数十秒間のサーバ電源 OFF 期間がありますが、自動で再起動されますので、電源ボタンによる電源 ON など行なわないようお願いします。

手動ロールバック実行手順はこれで終了となります。ロールバック結果は GUI を起動後、トレースログ画面で確認してください。

4. 手動ロールバックが正常終了した場合、アップデート実行中画面は消え、ロールバック完了ダイアログが表示されます。「OK」を選択するとデバイス情報画面に戻ります。



以上で、手動ロールバックは完了です。



手動ロールバックの実行状況の確認は、OS のイベントログで確認可能です。“Started all rollback.” が開始を示し、“Completed all rollback.” が完了を示します。イベントログの詳細は「Hitachi Server Navigator ユーザーズガイド アップデート機能」の「イベントログ一覧」を参照してください。



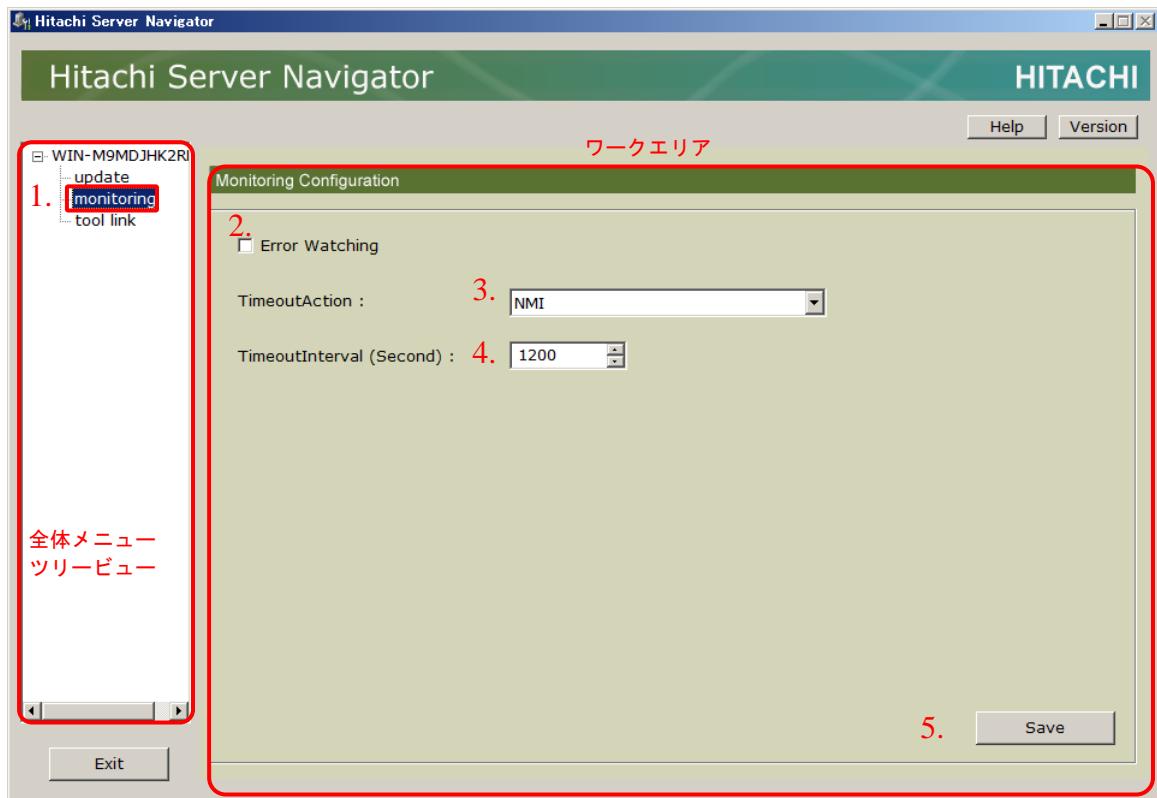
手動ロールバック実行中は、Server Navigator GUI を起動しないでください。ロールバック処理が異常終了する場合があります。



手動ロールバックが正常に行われたかの確認は、トレースログ画面で確認可能です。トレースログの内容については、「Hitachi Server Navigator ユーザーズガイド アップデート機能」の「トレースログの表示内容の詳細」を参照してください。

監視機能設定機能のGUI操作

監視機能設定機能の GUI 操作について詳細を説明します。

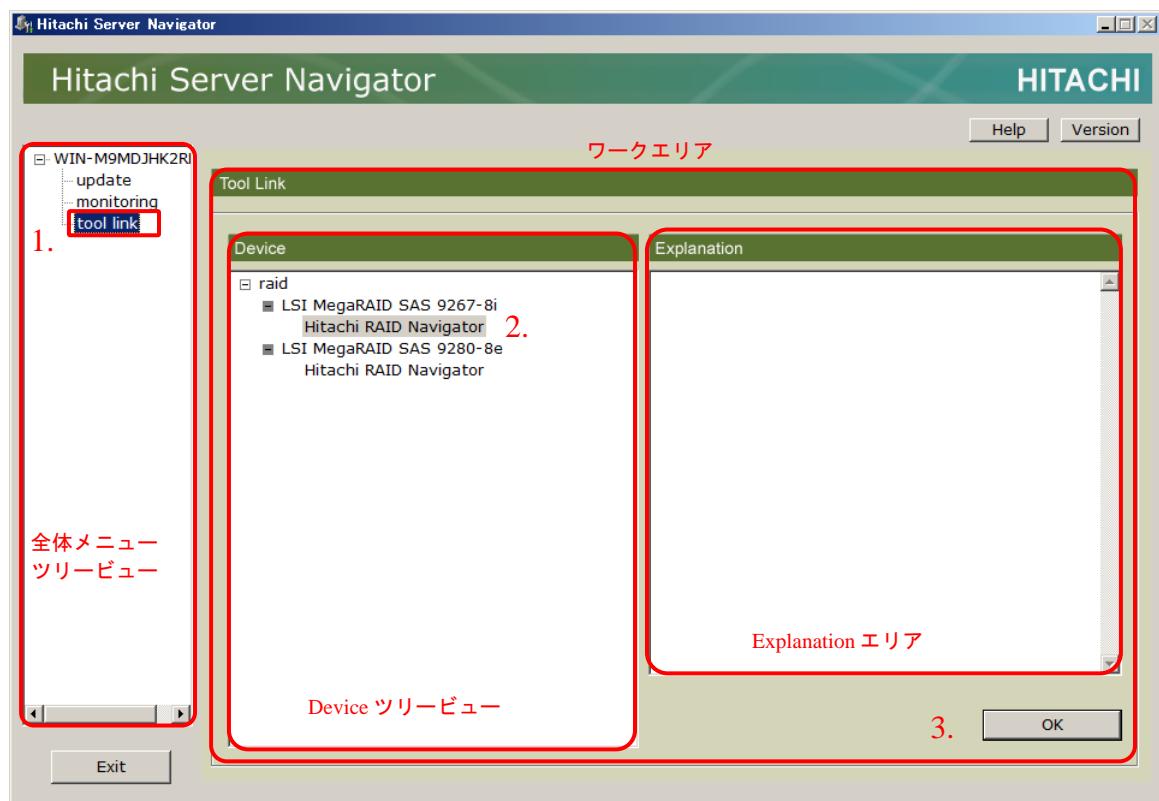


下記手順で、Server Navigator の Alive Monitor 機能の設定を行うことが可能です。

1. 全体メニューツリービューの「monitoring」をクリックして、ワークエリアを監視機能設定機能の GUI 画面に切り替えます。
2. Server Navigator の Alive Monitor 機能を有効にするには、ErrorWatching チェックボックスにチェックマークが付いている必要があります。チェックマークが付いていない場合は、チェックボックスをクリックしてチェックマークを付けます。
3. TimeoutActionリストボックスから、「[異常ログ監視でタイムアウト時に行うアクション](#)」(P.3-24)を選択します。
4. TimeoutInterval テキストボックスで、異常ログ監視のタイムアウトの間隔を設定します。
5. **OK** ボタンをクリックすると 2.~4.で変更した設定が保存されます。

ツール連携機能のGUI操作

ツール連携機能の GUI 操作について詳細を説明します。



下記手順で、Server Navigator に連携するツールを起動することができます。

1. 全体メニューツリービューの「tool link」をクリックして、ワークエリアをツール連携機能の GUI 画面に切り替えます。
2. Device ツリービューに表示されるツールから、起動したいツールをクリックして選択します。
3. **OK** ボタンをクリックすると 2.で選択したツールが起動されます。

Server NavigatorのCLI操作の説明

この章では、Server Navigator の GUI 操作を CLI での実行に置き換えた例について説明します。

- [Server NavigatorのCLI操作概要](#)
- [Server NavigatorのCLI操作例一覧](#)



この章の内容は、Server Navigator を使用してホストの運用・管理をするシステム管理者を対象としています。また、Windows、WMI、VBScript、Web について基本的な知識があることを前提としています。

Server NavigatorのCLI操作概要

Server Navigator の CLI での実行例の概要を以下に示します。

ローカルホストのCLI操作

本マニュアルで提供する CLI サンプルを引数なしで使用するとローカルホストで実行されます。 GUI 操作を手軽にバッチ実行に置き換えることが可能です。

リモートホストのCLI操作

本マニュアルで提供する CLI サンプルに引数として管理対象サーバのホスト名または IP アドレスを指定して実行すると、リモートホストである管理対象サーバ上の Server Navigator を操作することが可能です。

そのため管理対象サーバに Server Navigator がインストールされている必要がありますが、 CLI 実行元である管理サーバに Server Navigator がインストールされている必要はありません。

1 台の管理サーバから、複数台の管理対象サーバに対しリモートホストの CLI 操作を行うことにより、複数台の管理対象サーバを一括管理することが可能です。

WMI を使用してリモートホストの CLI 操作を行うため、安全に使用するにはファイアウォール設定に関する知識が必要となります。

Server NavigatorのCLI操作例一覧

Server Navigator の CLI での実行例の一覧を以下に示します。

- [デバイスリストの表示](#)
- [デバイス情報の表示](#)
- [アップデート情報の表示](#)
- [デバイスバージョンの表示](#)
- [ダウンロード指示](#)
- [アップデート指示](#)
- [ロールバック指示](#)

デバイスリストの表示

デバイスリストを表示する場合は下記サンプルを参照してください。

- [デバイスリスト表示\(CLI\)](#) (P.B-37)

サンプルソース `prtDevList.vbs` を実行すると、Server Navigator GUI のデバイス画面における Device List ツリービューの内容を全デバイス分採取し表示します。

デバイスリスト表示(CLI)のサンプルソースの実行例

```
C:\>cscript.exe //NoLogo prtDevList.vbs
System
  BIOS/EFI
  BMC
  raid
    LSI MegaRAID SAS 9267-8i
  ReturnValue GetProviderManageInfo: 0

C:\>
```

デバイス情報の表示

デバイス情報を表示する場合は下記サンプルを参照してください。

- [デバイス情報表示\(CLI\)](#) (P.B-44)

サンプルソース `prtDeviceInfo.vbs` を実行すると、Server Navigator GUI のデバイス画面における Device Info. エリアの内容を採取し表示します。

デバイス情報表示(CLI)のサンプルソースの実行例

```
C:\>cscript.exe //NoLogo prtDeviceInfo.vbs
System
  BIOS/EFI
    Description: HA8000/RS220-h
    Manufacturer: HITACHI
    Device ID: FFFFFFFF-FFFF-FFFF-FFFF-8C89A59C5A8C
  BMC
    Description: HA8000/RS220-h
    Manufacturer: HITACHI
    Device ID: FFFFFFFF-FFFF-FFFF-FFFF-8C89A59C5A8C
  raid
    LSI MegaRAID SAS 9267-8i
    Description: LSI MegaRAID SAS 9267-8i
    Manufacturer: LSI Corp.
    Device ID: 005B
    CtrlNo: ctrl#0
  ReturnValue GetProviderManageInfo: 0

C:\>
```

アップデート情報の表示

アップデート情報を表示する場合は下記サンプルを参照してください。

- [アップデート情報表示\(CLI\)](#) (P.B-49)

サンプルソース `prtUpdateInfo.vbs` を実行すると、Server Navigator GUI のデバイス画面における Update Info. エリアの内容を全デバイス分採取し表示します。

アップデート情報表示(CLI)のサンプルソースの実行例

```
C:¥>cscript.exe //NoLogo prtUpdateInfo.vbs
System
  BIOS/EFI
    [firm]
      Status: There is no update history.
      Module Name: rom_ep_103_kai.zip
      Current Version: MH.1.02
      Next Version: MH.1.03
      Priority: 4
      Reboot: True
      ReturnValue GetProviderManageInfo: 0
  BMC
    [firm]
      Status: There is no update history.
      Module Name: rom_ha8000xm.09.13.01.zip
      Current Version: 09-13
      ReturnValue GetProviderManageInfo: 0
  raid
    LSI MegaRAID SAS 9267-8i
      CtrlNo: ctrl#0
      [firm]
        Status: Application success
        Current Version: 3.151.05-1458
      [driver]
        Status: Application success
        Module Name: lsi_hwr_drv_w2k8x32_5.2.116.32.zip
        Current Version: 5.2.116.32
      [util]
        Status: Application success
        Module Name: msm_win_11.08.0302.zip
        Current Version: 11.08.03.0200
      ReturnValue GetProviderManageInfo: 0
C:¥>
```

本サンプルソースの実行例では以下のことが分かります。

- BIOS/EFI :

Status が "There is no update history." であり、アイドル状態。

現在の Firmware Version が "MH.1.02"。

Next Version があることから、アップデートモジュールをダウンロード完了し、アップデート可能な状態。

アップデートモジュールは、Version が "MH.1.03"、重要度 4 でシステムリブートが必要。

- BMC :

Status が "There is no update history." であり、アイドル状態。

現在の Firmware Version が "09-13"。

Next Version がないことから、アップデートがない。

- RAID :

Firmware、Driver、Utility の Status が何れも "Application success" であり、アップデート適用成功状態。

現在の Firmware Version が "3.151.05-1458"。

現在の Driver Version が "5.2.116.32"。

現在の Utility Version が "11.08.03.0200"。

Next Version がないことから、アップデートがない。

デバイスバージョンの表示

デバイスバージョンを表示する場合は下記サンプルを参照してください。

- [バージョン情報表示\(CLI\)](#) (P.B-58)

サンプルソース `prtDevVerInfo.vbs` を実行すると、Server Navigator GUI のデバイス画面における Update Info. エリアの内容を全デバイス分採取し、Current Version を表示します。

デバイスバージョン表示(CLI)のサンプルソースの実行例

```
C:\>cscript.exe //Nologo prtDevVerInfo.vbs
System
  BIOS/EFI
    [firm]
      Current Version: MH.1.03
  ReturnValue GetProviderManageInfo: 0
BMC
  [firm]
    Current Version: 09-12
  ReturnValue GetProviderManageInfo: 0
raid
  LSI MegaRAID SAS 9267-8i
    CtrlNo: ctrl#0
    [firm]
      Current Version: 3.152.85-1660
    [driver]
      Current Version: 5.2.112.64
    [util]
      Current Version: 1.2.0.18
  ReturnValue GetProviderManageInfo: 0
C:\>
```

ダウンロード指示

ダウンロード指示による手動ダウンロードを行う場合は下記サンプルを参照して順番に実行してください。



サンプルソース DownloadAll.vbs では、ダウンロード元に「外部媒体」を設定しています。ダウンロード元に「Web」を設定する場合は、サンプルソースの下記部分で 2 に設定しているパラメータを 0 に変更してください。

```
' ダウンロード元 0:web, 2:外部媒体
objSettingInParam.Properties_.Item("DownloadType") = 2
```

ダウンロード元に外部媒体を使用する場合

1. [ダウンロード指示\(CLI\)](#) (P.B-68)

サンプルソース DownloadAll.vbs では外部媒体パスに "C:\UpdateModule\01" を指定しています。御使用の環境に合わせ外部媒体パスを変更してください。サンプルソースを実行すると、手動ダウンロードを開始します。

2. [アップデート情報表示\(CLI\)](#) (P.B-49)

サンプルソース prtUpdateInfo.vbs を実行すると、ダウンロードの結果としてアップデートモジュールがあるか確認できます。

3. [トレースログ取得\(CLI\)](#) (P.B-90)

サンプルソース prtTraceLog.vbs を実行すると、ダウンロードの成否が確認できます。

ダウンロード元に Web を使用する場合

1. [自動ダウンロード設定更新\(CLI\)](#) (P.B-83)

サンプルソースの処理内容詳細を参考に、サンプルソース setDownloadSetting.vbs のダウンロード Web サイトに接続するための設定部分を変更してください。サンプルソースを実行すると、ダウンロードの設定を行います。

2. [ダウンロード指示\(CLI\)](#) (P.B-68)

サンプルソース DownloadAll.vbs のダウンロード元を「Web」に変更してください。サンプルソースを実行すると、手動ダウンロードを開始します。

3. [アップデート情報表示\(CLI\)](#) (P.B-49)

サンプルソース prtUpdateInfo.vbs を実行すると、ダウンロードの結果、アップデートモジュールがあるか確認できます。

4. [トレースログ取得\(CLI\)](#) (P.B-90)

サンプルソース prtTraceLog.vbs を実行すると、ダウンロードの成否が確認できます。

アップデート指示

アップデート指示による手動アップデートを行う場合は下記サンプルを参照して順番に実行してください。

1. [アップデート指示\(CLI\)](#) (P.B-68)

サンプルソース UpdateAll.vbs を実行すると、手動アップデートを開始します。

2. [アップデート進捗率表示\(CLI\)](#) (P.B-77)

サンプルソース prtUpdateProgress.vbs を実行すると、アップデートの進捗を確認できます。必要に応じ実行してください。

3. [アップデート情報表示\(CLI\)](#) (P.B-49)

アップデートが完了したらサンプルソース prtUpdateInfo.vbs を実行してください。アップデート結果が確認できます。

4. [トレースログ取得\(CLI\)](#) (P.B-90)

サンプルソース prtTraceLog.vbs を実行すると、アップデートの成否が確認できます。

ロールバック指示

ロールバック指示による手動ロールバックを行う場合は下記サンプルを参照して順番に実行してください。

1. [ロールバック指示\(CLI\)](#) (P.B-68)

サンプルソース RollbackAll.vbs を実行すると、手動ロールバックを開始します。

2. [アップデート進捗率表示\(CLI\)](#) (P.B-77)

サンプルソース prtUpdateProgress.vbs を実行すると、ロールバックの進捗を確認できます。必要に応じ実行してください。

3. [アップデート情報表示\(CLI\)](#) (P.B-49)

ロールバックが完了したらサンプルソース prtUpdateInfo.vbs を実行してください。ロールバック結果が確認できます。

4. [トレースログ取得\(CLI\)](#) (P.B-90)

サンプルソース prtTraceLog.vbs を実行すると、ロールバックの成否が確認できます。

トラブルシューティング

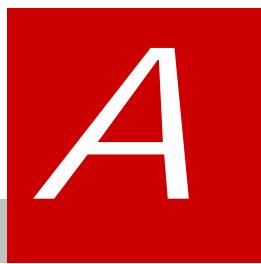
この章では、Server Navigator のトラブルシューティングについて説明します。

[トラブルシュートの概要](#)

トラブルシュートの概要

Server Navigator のトラブルシュートは、発生ケース毎に関連機能のドキュメントを参照して行ってください。

- Server Navigator のインストール/アンインストールで発生したメッセージやエラーの対処
[「インストールメッセージとファイル一覧」\(P.A-1\)](#)を参照してください。
- Server Navigator のアップデート機能または Server Navigator GUI/CLI で発生したメッセージやエラーの対処
[「Hitachi Server Navigator ユーザーズガイド アップデート機能」](#)を参照してください。
- Log Monitor 機能で発生したメッセージやエラーの対処
[「Hitachi Server Navigator ユーザーズガイド Log Monitor 機能」](#)
- Alive Monitor 機能で発生したメッセージやエラーの対処
[「Hitachi Server Navigator ユーザーズガイド Alive Monitor 機能」](#)
- RAID 管理機能で発生したメッセージやエラーの対処
[「Hitachi Server Navigator ユーザーズガイド RAID 管理機能」](#)



インストールメッセージとファイル一覧

この章では、メッセージ一覧とインストールファイル一覧について説明します。

- [インストール/アンインストール時のメッセージ一覧](#)
- [インストールファイル一覧](#)

インストール/アンインストール時のメッセージ一覧

Hitachi Server Navigator インストール/アンインストール時の出力メッセージと対処方法について、以下に示します。

サーバナビゲータから Hitachi Server Navigator をインストール/アンインストールする場合は、「サイレント版」の出力メッセージになります。

種類の「I」は Information(情報表示)であり、対処不要です。「W」は Warning(警告表示)であり不完全な状態なので、対処に従ってください。「E」は Error であり、対処が必要です。

種類	インストーラ		タイミング			出力 メッセージ	内容/対処
	サイレント版	通常版	Setup	Installation	Uninstallation		
I	○			○		The installation was successful. (Please eject the CDROM.)	インストールが正常終了しました。(DVDを取り出してください。)
I	○				○	The uninstallation was successful. (Please eject the CDROM.)	アンインストールが正常終了しました。(DVDを取り出してください。)
E	○			○		The installation failed. (Please eject the CDROM.)	インストールが異常終了しました。(DVDを取り出してください。) システム装置が正常動作しているか確認してください。(ハードウェアエラーが発生していないか)
E	○				○	The uninstallation failed. (Please eject the CDROM.)	アンインストールが異常終了しました。(DVDを取り出してください。) システム装置が正常動作しているか確認してください。(ハードウェアエラーが発生していないか)
E		○	○			The setup failed. (Please eject the CDROM.) This OS is not supported.	サポート外OSでインストーラを実行しました。(DVDを取り出してください。) 本システム装置環境が「 動作に必要なシステム環境 」(P.1-10)を満たしているかどうか確認してください。
E	○	○	○			The setup failed. (Please eject the CDROM.) DLL initialization error!	DLLの初期化(レジストリ登録)失敗によって、セットアップが異常終了しました。(DVDを取り出してください。) 本システム装置環境が「 動作に必要なシステム環境 」(P.1-10)を満たしているかどうか確認してください。

種類	インストーラ		タイミング			出力 メッセージ	内容/対処
	サイレント版	通常版	Setup	Installation	Uninstallation		
E	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>			The setup failed. (Please eject the CDROM.) A later version of the product is already installed on your computer. The setup cannot continue.	新しいバージョンがインストールされた状態で、セットアップを実行しました。(DVDを取り出してください。) ダウングレードはサポートしません。
E	<input type="radio"/>		<input type="radio"/>			The setup failed. (Please eject the CDROM.) Getting Version Information error!	バージョン情報取得失敗によって、セットアップが異常終了しました。(DVDを取り出してください。)
E	<input type="radio"/>	<input type="radio"/>		<input type="radio"/>		The installation failed. (Please eject the CDROM.) Backup error in configuration information!	引継ぎ情報保存失敗によって、インストールが異常終了しました。(DVDを取り出してください。) アンインストール後、再インストールしてください。
W	<input type="radio"/>	<input type="radio"/>		<input type="radio"/>		The installation was successful. (Please eject the CDROM.) However, it is necessary to set it again by the customer because it was not able to migrate it.	インストールは完了しました。(DVDを取り出してください。) ただし、引継ぎ情報設定、ダウンロード接続 WebSite の設定に失敗したため、再設定が必要です。
I		<input type="radio"/>	<input type="radio"/>			You must restart your computer after updating Hitachi Server Navigator. Click [Yes] to restart your computer now. Click [No] if you plan to restart later. *	セットアップが完了したため、再起動が必要。
I		<input type="radio"/>		<input type="radio"/>		Stop the Update ManagerService.	UpdateManagerServiceを停止します。
I	<input type="radio"/>	<input type="radio"/>			<input type="radio"/>	The uninstallation was successful. However, it is necessary to delete Server Navigator folder by the customer. (Please eject the CDROM.)	アンインストールが正常終了しました。(DVDを取り出してください。) 「Server Navigator」のフォルダは手動で削除をお願いします。
E		<input type="radio"/>	<input type="radio"/>			This machine is un-support model.	サポート外機種でインストーラを実行しました。 Virtage の LPAR(ゲスト OS)にはインストールできません。

* restart は、システムリブートを意味しています。

インストールファイル一覧

Server Navigator のインストールファイル一覧は以下の通りです。

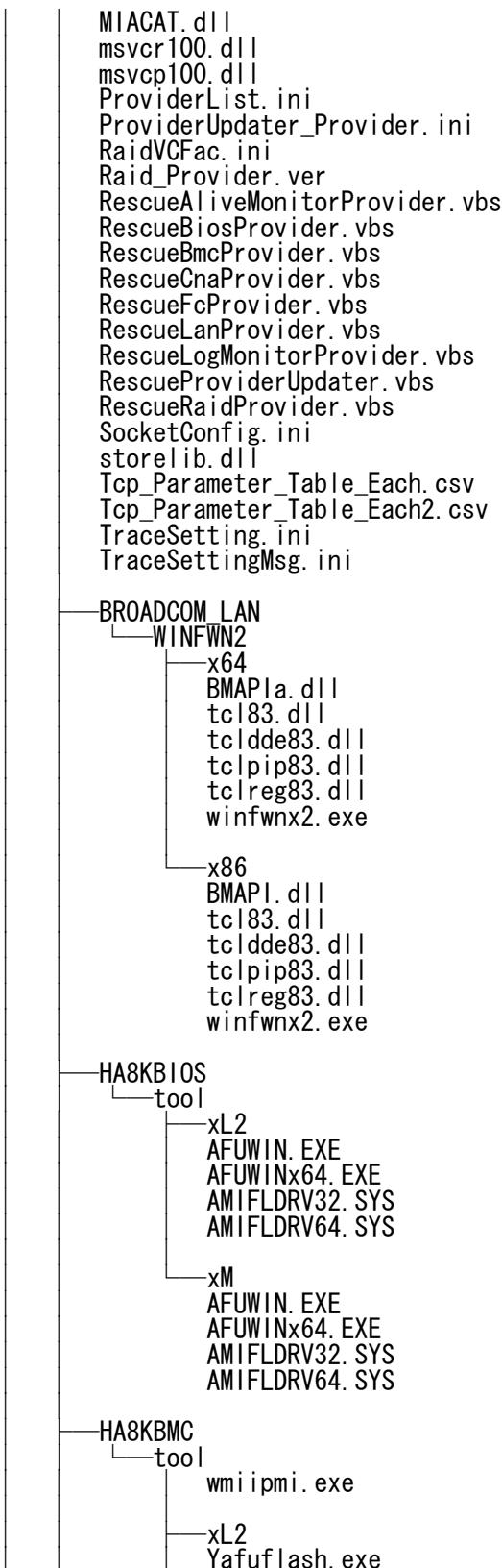
〈Server Navigator インストールフォルダ〉は、「[Windows版ServerNavigatorをインストールする場合](#)」(P.2-2)で設定したフォルダです。デフォルト設定は下記の通りです。

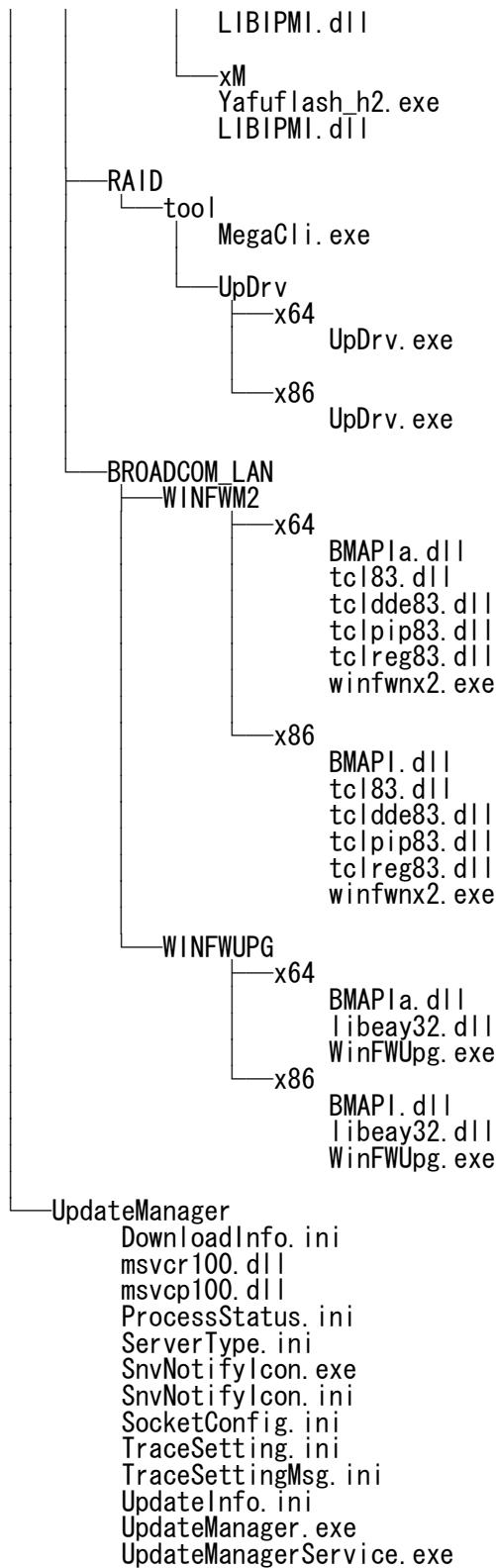
32 ビット版 OS : "C:\Program Files\Hitachi\Server Navigator"

64 ビット版 OS : "C:\Program Files (x86)\Hitachi\Server Navigator"

〈Server Navigator インストールフォルダ〉

GUI	ErrMsgSetting.ini GUISetting.ini Hitachi_GUI.exe license.txt msvcr100.dll setup.ico snv_usersguide.pdf TraceSetting.ini TraceSettingMsg.ini
Provider	AliveMonitor_Provider.ver BROADCOM_LAN.dll BiosVCFac.ini Bios_Provider.ver BmcVCFac.ini Bmc_Provider.ver cimhitachi.mof CnaVCFac.ini Cna_Provider.ver Driver_Parameter_Table.csv EMULEX_CNA.dll EMULEX_FC.dll Fc_Provider.ver FcVCFac.ini HITACHI_BIOS.dll HITACHI_BMC.dll Hitachi_DevServiceBios.dll Hitachi_DevServiceBmc.dll Hitachi_DevServiceCna.dll Hitachi_DevServiceFc.dll Hitachi_DevServiceLan.dll Hitachi_DevServiceRaid.dll HITACHI_FC.dll Hitachi_LogManager.dll Hitachi_ProviderUpdater.dll Hitachi_SnvVersion.dll Hitachi_ToolServiceLogMonitoring.dll Hitachi_ToolServiceWD.dll Hitachi_UpdateManager.dll Hitachi_VendorWD.dll Lan_Provider.ver LanVCFac.ini LogMonitor_Provider.ver LogManager_Provider.ini LSI_RAID.dll





<Windows フォルダ>
ServerNavigator.ini
ServerNavigatorEvent.dll

B

CLI 詳細

この付録では、Server Navigator の CLI の詳細について説明します。

- [Server Navigator CLI のメソッド一覧](#)
- [Server Navigator CLI で使用するクエリー一覧](#)
- [Server Navigator の CLI サンプル一覧](#)

Server Navigator CLI のメソッド一覧

Server Navigator の各機能を使用するため、OS 標準の WMI プロバイダの ExecMethod メソッドを使用して、Server Navigator の CIM Provider のメソッドを呼び出します。

Server Navigator の GUI 操作を CLI に置き換えるために使用する Server Navigator のクラスとメソッドについて説明します。

- [HITACHI_UpdateManager クラス](#)
- [HITACHI_LogManager クラス](#)
- [デバイス用CIM Provider用クラス](#)
- [各クラスのメソッド共通実行結果](#)

HITACHI_UpdateManager クラス

HITACHI_UpdateManager クラスで提供するメソッドについて説明します。

HITACHI_UpdateManager クラスのメソッド一覧

以下に HITACHI_UpdateManager クラスのメソッドの一覧を表記します。

メソッド	機能	説明
GetAutoDownloadSetting	自動ダウンロード設定 取得メソッド	Update Manager で管理している自動ダウンロードスケジュール情報と自動ダウンロード設定情報を取得する
SetAutoDownloadSetting	自動ダウンロード設定 更新メソッド	Update Manager で管理している自動ダウンロードスケジュール情報と自動ダウンロード設定情報を更新する
GetAutoUpdateSetting	自動アップデート設定 取得メソッド	Update Manager で管理している自動アップデートスケジュール情報を取得する
SetAutoUpdateSetting	自動アップデート設定 更新メソッド	Update Manager で管理している自動アップデートスケジュール情報を更新する
DownloadAll	ダウンロード指示メソッド	外部媒体あるいは web かキャッシュサーバからモジュールの手動ダウンロードを行うよう、Update Manager に指示する
UpdateAll	アップデート指示メソッド	選択したデバイスの手動アップデートを行うよう、Update Manager に指示する
RollbackAll	ロールバック指示メソッド	選択したデバイスの手動ロールバックを行うよう、Update Manager に指示する
ExecuteReboot	システムリブート指示 メソッド	BMC 用 CIM Provider に対してシステムリブートを実行する よう、Update Manager に指示する

GetAutoDownloadSetting メソッド

以下に GetAutoDownloadSetting メソッドを使用するサンプルソースを示します。

- [ダウンロード指示のサンプルソース](#)
- [自動ダウンロード設定取得のサンプルソース](#)

GetAutoDownloadSetting メソッドの引数

GetAutoDownloadSetting メソッドに引数はありません。

GetAutoDownloadSetting メソッドの戻り値

以下に GetAutoDownloadSetting メソッドの戻り値を表記します。

戻り値名	項目	型	説明
DownloadSchedule	ダウンロードスケジュール	Hitachi_DownloadSchedule	ダウンロードスケジュールパラメータセット
IsAutoSetting	自動更新	bool	自動ダウンロードの可否 True: 自動ダウンロードを行う、False: 自動ダウンロードを行わない
RepeatType	繰り返し	int	自動ダウンロードの繰り返しの可否 0: 日付指定・繰り返しなし 1: 曜日指定・繰り返しなし
Date	日付	string	自動ダウンロード実行日付 "01"~"31"の間の値 "01": 1 日、"25": 25 日
DayOfWeek	曜日	string	自動ダウンロード(繰り返し)実行曜日 "0": 日曜日、"1": 月曜日、"2": 火曜日、 "3": 水曜日、"4": 木曜日、"5": 金曜日、 "6": 土曜日
Time	時刻	string	自動ダウンロード実行時刻 "0000"~"2359"の間の値 "0800": 8 時 00 分 "2230": 22 時 30 分
DownloadSetting	ダウンロード設定	Hitachi_DownloadSetting	ダウンロード設定パラメータセット
DownloadType	ダウンロード元	int	自動ダウンロード時の接続先 0: WebSite 2: 外部媒体(ダウンロード指示時のみ有効)
WebSiteUrl	URL	string	WebSite の URL
ProxyAddress	プロキシ	string	プロキシサーバのアドレス
ProxyPort	ポート	string	プロキシサーバのポート番号
ProxyUser	プロキシユーザ	string	プロキシサーバのユーザ情報

戻り値名	項目	型	説明
ProxyPassword	プロキシパスワード	string	プロキシサーバのパスワード
CacheServerPath	キャッシュサーバパス	string	キャッシュサーバのパス
CacheServerUser	キャッシュサーバユーザ	string	キャッシュサーバのユーザ情報
CacheServerPassword	キャッシュサーバパスワード	string	キャッシュサーバのパスワード
TempDriveLetter	一時接続ドライブレター	string	キャッシュサーバからのダウンロード時に一時的に割り付けるネットワークドライブレター A: ~ "Z: "までの間の値 "A: ": A ドライブ、"N: ": N ドライブ
ExternalMediaPath	外部媒体パス	string	モジュールを格納した外部媒体のパス
ReturnValue	処理結果	long	メソッド実行結果

SetAutoDownloadSetting メソッド

以下に SetAutoDownloadSetting メソッドを使用するサンプルソースを示します。

- [自動ダウンロード設定更新のサンプルソース](#)

SetAutoDownloadSetting メソッドの引数

以下に SetAutoDownloadSetting メソッドの引数を表記します。

引数名	項目	型	説明	引数の設定要否
DownloadSchedule	ダウンロードスケジュール	Hitachi_DownloadSchedule	ダウンロードスケジュールパラメータセット	必須
IsAutoSetting	自動更新	bool	自動ダウンロードの可否 True: 自動ダウンロードを行う、False: 自動ダウンロードを行わない	必須
RepeatType	繰り返し	int	自動ダウンロードの繰り返しの可否 0: 日付指定・繰り返しなし 1: 曜日指定・繰り返しなし	必須
Date	日付	string	自動ダウンロード実行日付 "01"~"31"の間の値 "01": 1 日、"25": 25 日	必須
DayOfWeek	曜日	string	自動ダウンロード(繰り返し)実行曜日 "0": 日曜日、"1": 月曜日、"2": 火曜日、 "3": 水曜日、"4": 木曜日、"5": 金曜日、 "6": 土曜日	必須

引数名	項目	型	説明	引数の設定要否
Time	時刻	string	自動ダウンロード実行時刻 "0000"～"2359"の間の値 "0800": 8 時 00 分 "2230": 22 時 30 分	必須
DownloadSetting	ダウンロード設定	Hitachi_DownloadSetting	ダウンロード設定パラメータセット	必須
	DownloadType	int	自動ダウンロード時の接続先 0: WebSite 2: 外部媒体(ダウンロード指示時のみ有効)	必須
	WebSiteUrl	URL	WebSite の URL	web からダウンロードする場合必須
	ProxyAddress	string	プロキシサーバのアドレス	プロキシを使用する場合必須
	ProxyPort	string	プロキシサーバのポート番号	プロキシを使用する場合必須
	ProxyUser	string	プロキシサーバのユーザ情報	プロキシを使用する場合必須
	ProxyPassword	string	プロキシサーバのパスワード	プロキシを使用する場合必須
	CacheServerPath	string	キャッシュサーバのパス	キャッシュサーバからダウンロードする場合必須
	CacheServerUser	string	キャッシュサーバのユーザ情報	キャッシュサーバ接続に認証が必要の場合必須
	CacheServerPassword	string	キャッシュサーバのパスワード	キャッシュサーバ接続に認証が必要の場合必須
TempDriveLetter	一時接続ドライブレター	string	キャッシュサーバからのダウンロード時に一時的に割り付けるネットワークドライブレター A:～"Z:"までの間の値 "A:":A ドライブ、"N:":N ドライブ	キャッシュサーバからダウンロードする場合必須
ExternalMediaPath	外部媒体パス	string	モジュールを格納した外部媒体のパス	不要

SetAutoDownloadSetting メソッドの戻り値

以下に SetAutoDownloadSetting メソッドの戻り値を表記します。

戻り値名	項目	型	説明
ReturnValue	処理結果	long	メソッド実行結果

GetAutoUpdateSetting メソッド

以下に GetAutoUpdateSetting メソッドを使用するサンプルソースを示します。

- [自動アップデート設定取得のサンプルソース](#)

GetAutoUpdateSetting メソッドの引数

GetAutoUpdateSetting メソッドに引数はありません。

GetAutoUpdateSetting メソッドの戻り値

以下に GetAutoUpdateSetting メソッドの戻り値を表記します。

戻り値名	項目	型	説明
UpdateSchedule	アップデートスケジュール	Hitachi_UpdateSchedule	アップデートスケジュールパラメータセット
IsAutoSetting	自動更新	bool	自動アップデートの可否 True: 自動アップデートを行う、False: 自動アップデートを行わない
RepeatType	繰り返し	int	自動アップデートの繰り返しの可否 0: 日付指定・繰り返しなし 1: 曜日指定・繰り返しなし
Date	日付	string	自動アップデート実行日付 "01"~"31"の間の値 "01": 1 日、"25": 25 日
DayOfWeek	曜日	string	自動アップデート(繰り返し)実行曜日 "0": 日曜日、"1": 月曜日、"2": 火曜日、 "3": 水曜日、"4": 木曜日、"5": 金曜日、 "6": 土曜日
Time	時刻	string	自動アップデート実行時刻 "0000"~"2359"の間の値 "0800": 8 時 00 分 "2230": 22 時 30 分
ReturnValue	処理結果	long	メソッド実行結果

SetAutoUpdateSetting メソッド

以下に SetAutoUpdateSetting メソッドを使用するサンプルソースを示します。

- [自動アップデート設定更新のサンプルソース](#)

SetAutoUpdateSetting メソッドの引数

以下に SetAutoUpdateSetting メソッドの引数を表記します。

引数名	項目	型	説明	引数の設定要否
UpdateSchedule	アップデートスケジュール	Hitachi_UpdateSchedule	アップデートスケジュールパラメータセット	必須
IsAutoSetting	自動更新	bool	自動アップデートの可否 True: 自動アップデートを行う、False: 自動アップデートを行わない	必須
RepeatType	繰り返し	int	自動アップデートの繰り返しの可否 0: 日付指定・繰り返しなし 1: 曜日指定・繰り返しなし	必須
Date	日付	string	自動アップデート実行日付 "01"~"31"の間の値 "01": 1 日、"25": 25 日	必須
DayOfWeek	曜日	string	自動アップデート(繰り返し)実行曜日 "0": 日曜日、"1": 月曜日、"2": 火曜日、 "3": 水曜日、"4": 木曜日、"5": 金曜日、 "6": 土曜日	必須
Time	時刻	string	自動アップデート実行時刻 "0000"~"2359"の間の値 "0800": 8 時 00 分 "2230": 22 時 30 分	必須

SetAutoUpdateSetting メソッドの戻り値

以下に SetAutoUpdateSetting メソッドの戻り値を表記します。

戻り値名	項目	型	説明
ReturnValue	処理結果	long	メソッド実行結果

DownloadAll メソッド

以下に DownloadAll メソッドを使用するサンプルソースを示します。

- [ダウンロード指示のサンプルソース](#)

DownloadAll メソッドの引数

以下に DownloadAll メソッドの引数を表記します。

引数名	項目	型	説明	引数の設定要否
DownloadSetting	ダウンロード設定	Hitachi_DownloadSetting	ダウンロード設定パラメータセット	必須
DownloadType	ダウンロード元	int	自動ダウンロード時の接続先 0: WebSite 2: 外部媒体(ダウンロード指示時のみ有効)	必須
WebSiteUrl	URL	string	WebSite の URL	web からダウンロードする場合必須
ProxyAddress	プロキシ	string	プロキシサーバのアドレス	任意
ProxyPort	ポート	String	プロキシサーバのポート番号	任意
ProxyUser	プロキシユーザ	String	プロキシサーバのユーザ情報	任意
ProxyPassword	プロキシパスワード	String	プロキシサーバのパスワード	任意
CacheServerPath	キャッシュサーバパス	string	キャッシュサーバのパス	キャッシュサーバからダウンロードする場合必須
CacheServerUser	キャッシュサーバユーザ	string	キャッシュサーバのユーザ情報	キャッシュサーバ接続に認証が必要の場合必須
CacheServerPassword	キャッシュサーバパスワード	string	キャッシュサーバのパスワード	キャッシュサーバ接続に認証が必要の場合必須
TempDriveLetter	一時接続ドライブレター	string	キャッシュサーバからのダウンロード時に一時的に割り付けるネットワークドライブレター A: ~ "Z: "までの間の値 "A: ": A ドライブ、"N: ": N ドライブ	キャッシュサーバからダウンロードする場合必須
ExternalMediaPath	外部媒体パス	String	モジュールを格納した外部媒体のパス	外部媒体からダウンロードする場合必須

DownloadAll メソッドの戻り値

以下に DownloadAll メソッドの戻り値を表記します。

戻り値名	項目	型	説明
ReturnValue	処理結果	long	メソッド実行結果

UpdateAll メソッド

以下に UpdateAll メソッドを使用するサンプルソースを示します。

- [アップデート指示のサンプルソース](#)

UpdateAll メソッドの引数

UpdateAll メソッドに引数はありません。

UpdateAll メソッドの戻り値

以下に UpdateAll メソッドの戻り値を表記します。

戻り値名	項目	型	説明
ReturnValue	処理結果	long	メソッド実行結果
Reboot	リブート要否	long	リブート要否 0: リブート不要 21: リブート誘導 22: 強制リブート 23: 強制シャットダウン

RollbackAll メソッド

以下に RollbackAll メソッドを使用するサンプルソースを示します。

- [ロールバック指示のサンプルソース](#)

RollbackAll メソッドの引数

RollbackAll メソッドに引数はありません。

RollbackAll メソッドの戻り値

以下に RollbackAll メソッドの戻り値を表記します。

戻り値名	項目	型	説明
ReturnValue	処理結果	long	メソッド実行結果
Reboot	リブート要否	long	リブート要否 0: リブート不要 21: リブート誘導 22: 強制リブート 23: 強制シャットダウン

ExecuteReboot メソッド

以下に ExecuteReboot メソッドを使用するサンプルソースを示します。

- [アップデート指示のサンプルソース](#)
- [ロールバック指示のサンプルソース](#)

ExecuteReboot メソッドの引数

ExecuteReboot メソッドに引数はありません。

ExecuteReboot メソッドの戻り値

以下に ExecuteReboot メソッドの戻り値を表記します。

戻り値名	項目	型	説明
ReturnValue	処理結果	long	メソッド実行結果

HITACHI_LogManagerクラス

HITACHI_LogManager クラスで提供するメソッドについて説明します。

HITACHI_LogManager クラスのメソッド一覧

以下に HITACHI_LogManager クラスのメソッドの一覧を表記します。

メソッド	機能	説明
CollectResultLog	処理結果ログ収集メソッド	Update Manager や CIM Provider が output したトレースログを収集する
CollectVersionUpLog	アップデート履歴収集メソッド	CIM Provider が output したトレースログからアップデート履歴ログを検索・収集し、「プロバイダ名」「日時」「デバイス特定情報(ベンダ名+ユニーク名(+固有情報))」「モジュール種別」「アップデート前バージョン」「アップデート後バージョン」の項目に分解して取得する

CollectResultLog メソッド

以下に CollectResultLog メソッドを使用するサンプルソースを示します。

- [トレースログ取得のサンプルソース](#)

CollectResultLog メソッドの引数

以下に CollectResultLog メソッドの引数を表記します。

引数名	項目	型	説明	引数の設定要否
FromDateTime	ログ収集開始日時	string	ログの収集を開始する日時 yyyy/mm/dd hh:mm:ss	任意

CollectResultLog メソッドの戻り値

以下に CollectResultLog メソッドの戻り値を表記します。

戻り値名	項目	型	説明
LogList	ログリスト	list<string>	ログ出力日時の降順にソートした文字列配列
ReturnValue	処理結果	long	メソッド実行結果

CollectVersionUpLog メソッド

以下に CollectVersionUpLog メソッドを使用するサンプルソースを示します。

- [アップデート履歴取得のサンプルソース](#)

CollectVersionUpLog メソッドの引数

CollectVersionUpLog メソッドに引数はありません。

CollectVersionUpLog メソッドの戻り値

以下に CollectVersionUpLog メソッドの戻り値を表記します。

戻り値名	項目	型	説明
VersionUpdateLogList	アップデート履歴ログリスト	list<Hitachi_VersionUpdateLog>	アップデート履歴ログリスト
ProviderName	プロバイダ名	string	プロバイダ名
DateTime	日時	string	アップデート履歴ログの出力日時
VendorName	ベンダ名	string	ベンダ名
UniqueName	ユニーク名	string	ユニーク名
ModuleKind	モジュール種別	string	モジュール種別 "util":ユティリティ
BeforeVersion	アップデート前バージョン	int	アップデートを行う前のベンダバージョン
AfterVersion	アップデート後バージョン	string	アップデートを行った後のベンダバージョン
ReturnValue	処理結果	long	メソッド実行結果

HITACHI_SnvVersion クラス

HITACHI_SnvVersion クラスで提供するメソッドについて説明します。

HITACHI_SnvVersion クラスのメソッド一覧

以下に HITACHI_SnvVersion クラスのメソッドの一覧を表記します。

メソッド	機能	説明
CollectSnvVersion	バージョン収集メソッド	レジストリに登録されている統合バージョンと、SNV のモジュール(exe, dll)のファイルバージョンを取得する

CollectSnvVersion メソッド

以下に CollectSnvVersion メソッドを使用するサンプルソースを示します。

- [バージョン情報表示のサンプルソース](#)

CollectSnvVersion メソッドの引数

CollectSnvVersion メソッドに引数はありません。

CollectSnvVersion メソッドの戻り値

以下に CollectSnvVersion メソッドの戻り値を表記します。

戻り値名	項目	型	説明
SnvVersionInfoList	バージョン情報リスト	list<Hitachi_SnvVersionInfo>	バージョン情報リスト
GroupName	グループ名	string	SNV のモジュールグループ名 "Product Version": 統合バージョン、 "GUI": GUI、 "Update Manager": UpdateManager、 "CIM-Provider": CIM Provider
FileName	ファイル名	string	SNV のモジュールファイル名 統合バージョンの場合は"-"
FileVersion	ファイルバージョン	string	SNV のモジュールファイルバージョン
ReturnValue	処理結果	long	メソッド実行結果

デバイスおよびツール用CIM Provider用クラス

デバイスおよびツール用 CIM Provider 用クラスで提供するメソッドについて説明します。

デバイス用 CIM Provider 用クラス一覧

以下にデバイス用 CIM Provider 用クラスの一覧を表記します。

CIM Provider	クラス
BIOS/EFI 用 CIM Provider	HITACHI_DevServiceBios
BMC 用 CIM Provider	HITACHI_DevServiceBMC
RAID 用 CIM Provider	HITACHI_DevServiceRaid
CNA 用 CIM Provider	HITACHI_DevServiceCna
FC 用 CIM Provider	HITACHI_DevServiceFc
LAN 用 CIM Provider	HITACHI_DevServiceLan

ツール用 CIM Provider 用クラス一覧

以下にツール用 CIM Provider 用クラスの一覧を表記します。

CIM Provider	クラス
ログ監視 CIM Provider	HITACHI_ToolServiceLogMonitoring
監視 CIM Provider	HITACHI_ToolServiceWD

デバイスおよびツール用 CIM Provider 用クラスのメソッド一覧

以下にデバイスおよびツール用 CIM Provider 用クラスのメソッドの一覧を表記します。

メソッド	機能	説明
GetProviderManageInfo (デバイス用CIM Provider)	プロバイダ管理情報取得メソッド (デバイス用 CIM Provider)	デバイス用 CIM Provider が管理している、プロバイダに関する情報やデバイス用モジュールの情報、リポジトリ情報を取得する
GetProviderManageInfo (ツール用CIM Provider)	プロバイダ管理情報取得メソッド (ツール用 CIM Provider)	ツール用 CIM Provider が管理している、プロバイダに関する情報やツール用モジュールの情報、リポジトリ情報を取得する
GetOperatingAuthority	操作権限取得メソッド	デバイスおよびツール用 CIM Provider の操作権限を取得する
ReleaseOperatingAuthority	操作権限解放メソッド	デバイスおよびツール用 CIM Provider の操作権限を解放する
SetUpdatePriority (デバイス用CIM Provider)	更新適用重要度設定メソッド (デバイス用 CIM Provider)	ダウンロードおよびアップデート時に参照する重要度を設定する
SetUpdatePriority (ツール用CIM Provider)	更新適用重要度設定メソッド (ツール用 CIM Provider)	ダウンロードおよびアップデート時に参照する重要度を設定する

メソッド	機能	説明
<u>GetUpdatePlans</u> (デバイス用CIM Provider)	アップデート予定リスト取得 メソッド (デバイス用 CIM Provider)	デバイス用 CIM Provider が管理するデバイス群のモジュール毎にアップデートが実行可能か確認し、確認した結果を取得する
<u>GetUpdatePlans</u> (ツール用CIM Provider)	アップデート予定リスト取得 メソッド (ツール用 CIM Provider)	ツール用 CIM Provider が管理するデバイス群のモジュール毎にアップデートが実行可能か確認し、確認した結果を取得する

GetProviderManageInfo メソッド

以下に GetProviderManageInfo メソッドを使用するサンプルソースを示します。

- [デバイスリスト表示のサンプルソース](#)
- [デバイス情報表示のサンプルソース](#)
- [アップデート情報表示のサンプルソース](#)
- [troubleshoot表示のサンプルソース](#)
- [Release Note表示のサンプルソース](#)
- [アップデート進捗率表示のサンプルソース](#)
- [更新適用重要度表示のサンプルソース](#)
- [更新適用重要度設定のサンプルソース 1](#)
- [更新適用重要度設定のサンプルソース 2](#)
- [アップデート予定リスト取得のサンプルソース](#)
- [連携ツール表示のサンプルソース](#)

GetProviderManageInfo メソッドの引数

GetProviderManageInfo メソッドに引数はありません。

GetProviderManageInfo メソッド(デバイス用 CIM Provider)の戻り値

以下に GetProviderManageInfo メソッド(デバイス用 CIM Provider)の戻り値を表記します。

戻り値名	項目	型	説明
DevProviderManageInfo	プロバイダ管理情報	Hitachi_DevProviderManageInfo	プロバイダに関する情報
ProviderName	string	string	プロバイダ名
ProviderVendorName	string	string	プロバイダを作成したベンダ名
ProviderVersion	string	string	プロバイダのバージョン

戻り値名	項目	型	説明
DeviceInfo	接続デバイス情報	list<Hitachi_DeviceInfo>	接続しているデバイスに関する情報 接続数分、情報を繰り返す
VendorName	ベンダ名	string	デバイスのベンダ名
UniqueName	ユニーク名	string	更新モジュールを特定するためのデバイス名
DeviceId	デバイス ID	string	接続しているデバイスを一意に認識する ID
HardwareVersion	稼動ハードバージョン	string	接続しているデバイスのハードバージョン
ExtendedInfo	拡張情報	list<string>	デバイスが任意で出力する情報 キーと値を組み合わせた文字列の配列で、セパレータは"?"
LinkAppInfo	アプリケーション情報	list<Hitachi_LinkApplicationInfo>	デバイスに関するアプリケーション情報
ApplicationName	アプリケーション名称	string	アプリケーションの名称
ExecutePath	実行パス	string	起動するための実行ファイルのフルパス
DescriptionFilePath	説明文テキストパス	string	アプリケーションの説明を記述したテキストのフルパス
ModuleInfo	モジュール情報	list<Hitachi_ManageModuleInfo>	管理対象のモジュール情報 モジュール種別分、情報を繰り返す
ModuleKind	モジュール種別	string	モジュール種別 "firm": フームウェア、"driver": ドライバ、"util": ユーティリティ
ModuleName	モジュール名	string	プログラム内でのモジュールを一意にする名称 特殊な util のみ名称を保有
UpdatePriority	更新適用重要度	int	更新モジュールを適用するレベル 1~5 の間の値、9: 適用しない
UpdateStatus	アップデートステータス	int	最新のアップデート処理に対しての状態ステータス
UpdateProgress	アップデート進捗率	int	アップデート機能の実行状態を進捗率で表す
RepositoryInfo	リポジトリ管理モジュール情報	list<Hitachi_ManageRepositoryInfo>	リポジトリに管理されているモジュール情報 管理している世代数分、情報を繰り返す
ApplyKind	適用種別	string	リポジトリの適用種別 "current": 稼動バージョン、 "update": 次回適用バージョン、"old": 旧稼動バージョン

戻り値名		項目	型	説明
		VendorVersion	string	ベンダバージョン
		DisplayName	string	モジュールの表示名称
		ModulePath	string	モジュールパス
		Priority	int	適用重要度 1~5 の間の値
		ReqProviderVersion	string	このモジュールのアップデートができるプロバイダバージョン
		RebootAfterPreUpdate	string	アップデート前処理後のリブート要否 "none": リブート不要、 "message": リブートを誘導 "force": 完了時に強制リブート、 "shutdown": 完了時に強制シャットダウン
		RebootAfterUpdate	string	アップデート後のリブート要否 "none": リブート不要、 "message": リブートを誘導 "force": 完了時に強制リブート、 "shutdown": 完了時に強制シャットダウン
		RebootAfterRestore	string	リストア後のリブート要否 "none": リブート不要、 "message": リブートを誘導 "force": 完了時に強制リブート、 "shutdown": 完了時に強制シャットダウン
		Recovery	bool	設定リカバリ要否 True: リカバリ要、False: リカバリ不要
		ModuleFileName	list<string>	モジュールパスからの絶対ファイルパス
		ManualFileName	list<string>	モジュールパスからの絶対ファイルパス
		ReleasenotesFileName	list<string>	モジュールパスからの絶対ファイルパス
		DependencyInfoSelf	list<Hitachi_Dependency_Info>	複数存在する場合、情報を繰り返す
		Provider Name	string	自プロバイダ名 自プロバイダのため値の設定なし
		VendorName	string	自プロバイダのベンダ名 自プロバイダのため値の設定なし
		UniqueName	string	自プロバイダのユニーク名 自プロバイダのため値の設定なし

戻り値名		項目	型	説明
DependencyInfoOther	ModuleKind	モジュール種別	string	モジュール種別 "firm": ファームウェア、 "driver": ドライバ、 "util": ユーティリティ
	ModuleName	モジュール名	string	プログラム内でのモジュールを一意にする名称 特殊な util のみ名称を保有
	VendorVersion	ベンダバージョン	string	下限ベンダバージョン
	UpperVendorVersion	上限ベンダバージョン	string	上限ベンダバージョン
	DependencyInfoOther	他デバイス依存情報	list<Hitachi_DeviceDependencyInfo>	複数存在する場合、情報を繰り返す
	Provider Name	プロバイダ名	string	他プロバイダ名
	VendorName	ベンダ名	string	他プロバイダのベンダ名
	UniqueName	ユニーク名	string	他プロバイダのユニーク名
	ModuleKind	モジュール種別	string	モジュール種別 "firm": ファームウェア、 "driver": ドライバ、 "util": ユーティリティ
	ModuleName	モジュール名	string	プログラム内でのモジュールを一意にする名称 特殊な util のみ名称を保有
ReturnValue	VendorVersion	ベンダバージョン	string	下限ベンダバージョン
	UpperVendorVersion	上限ベンダバージョン	string	上限ベンダバージョン
ReturnValue		処理結果	long	メソッド実行結果

GetProviderManageInfo メソッド(ツール用 CIM Provider)の戻り値

以下に GetProviderManageInfo メソッド(ツール用 CIM Provider)の戻り値を表記します。

戻り値名	項目	型	説明
DevProviderManageInfo	プロバイダ管理情報	Hitachi_Dev_ProviderManageInfo	プロバイダに関する情報
ProviderName	string	string	プロバイダ名
ProviderVendorName	string	string	プロバイダを作成したベンダ名
ProviderVersion	string	string	プロバイダのバージョン
ToolInfo	接続ツール情報	list<Hitachi_ToolInfo>	接続しているツールに関する情報 接続数分、情報を繰り返す

戻り値名	項目	型	説明
VendorName	ベンダ名	string	デバイスのベンダ名
UniqueName	ユニーク名	string	更新モジュールを特定するためのデバイス名
ExtendedInfo	拡張情報	list<string>	ツールが任意で出力する情報 キーと値を組み合わせた文字列の配列で、セパレータは"?"
LinkAppInfo	アプリケーション情報	list<Hitachi_LinkApplicationInfo>	ツールに関するアプリケーション情報
ApplicationName	アプリケーション名称	string	アプリケーションの名称
ExecutePath	実行パス	string	起動するための実行ファイルのフルパス
DescriptionFilePath	説明文テキストパス	string	アプリケーションの説明を記述したテキストのフルパス
ModuleInfo	モジュール情報	list<Hitachi_ManageModuleInfo>	管理対象のモジュール情報 モジュール種別分、情報を繰り返す
ModuleKind	モジュール種別	string	モジュール種別 "util":ユーティリティ
ModuleName	モジュール名	string	プログラム内でのモジュールを一意にする名称 特殊な util のみ名称を保有
UpdatePriority	更新適用重要度	int	更新モジュールを適用するレベル 1~5 の間の値、9:適用しない
UpdateStatus	アップデートステータス	int	最新のアップデート処理に対しての状態ステータス
UpdateProgress	アップデート進捗率	int	アップデート機能の実行状態を進捗率で表す
RepositoryInfo	リポジトリ管理モジュール情報	list<Hitachi_ManageRepositoryInfo>	リポジトリに管理されているモジュール情報 管理している世代数分、情報を繰り返す
ApplyKind	適用種別	string	リポジトリの適用種別 "current":稼動バージョン、 "update":次回適用バージョン、"old": 旧稼動バージョン
VendorVersion	ベンダバージョン	string	ベンダバージョン
DisplayName	表示名称	string	モジュールの表示名称
ModulePath	モジュールパス	string	モジュール格納パス
Priority	重要度	int	適用重要度 1~5 の間の値

戻り値名		項目	型	説明
	ReqProviderVersion	必要プロバイダバージョン	string	このモジュールのアップデートができるプロバイダバージョン
	RebootAfterPreUpdate	アップデート前処理後リブート要否	string	アップデート前処理後のリブート要否 "none": リブート不要、 "message": リブートを誘導 "force": 完了時に強制リブート、 "shutdown": 完了時に強制シャットダウン
	RebootAfterUpdate	アップデート後リブート要否	string	アップデート後のリブート要否 "none": リブート不要、 "message": リブートを誘導 "force": 完了時に強制リブート、 "shutdown": 完了時に強制シャットダウン
	RebootAfterRestore	リストア後リブート要否	string	リストア後のリブート要否 "none": リブート不要、 "message": リブートを誘導 "force": 完了時に強制リブート、 "shutdown": 完了時に強制シャットダウン
	Recovery	設定リカバリ要否	bool	設定リカバリ要否 True: リカバリ要、False: リカバリ不要
	ModuleFileName	更新モジュールファイル名	list<string>	モジュールパスからの絶対ファイルパス
	ManualFileName	マニュアルファイル名	list<string>	モジュールパスからの絶対ファイルパス
	ReleasenotesFileName	リリースノートファイル名	list<string>	モジュールパスからの絶対ファイルパス
	DependencyInfoSelf	プロバイダ内依存情報	list<Hitachi_Dependency_Info>	複数存在する場合、情報を繰り返す
	Provider Name	プロバイダ名	string	自プロバイダ名 自プロバイダのため値の設定なし
	VendorName	ベンダ名	string	自プロバイダのベンダ名 自プロバイダのため値の設定なし
	UniqueName	ユニーク名	string	自プロバイダのユニーク名 自プロバイダのため値の設定なし
	ModuleKind	モジュール種別	string	モジュール種別 "firm": ファームウェア、 "driver": ドライバ、 "util": ユーティリティ
	ModuleName	モジュール名	string	プログラム内でのモジュールを一意にする名称 特殊な util のみ名称を保有

戻り値名		項目	型	説明
DependencyInfoOther	VendorVersion	ベンダバージョン	string	下限ベンダバージョン
	UpperVendorVersion	上限ベンダバージョン	string	上限ベンダバージョン
	DependencyInfoOther	他デバイス依存情報	list<Hitachi_DeviceDependencyInfo>	複数存在する場合、情報を繰り返す
	Provider Name	プロバイダ名	string	他プロバイダ名
	VendorName	ベンダ名	string	他プロバイダのベンダ名
	UniqueName	ユニーク名	string	他プロバイダのユニーク名
	ModuleKind	モジュール種別	string	モジュール種別 "firm": フームウェア、 "driver": ドライバ、 "util": ユーティリティ
	ModuleName	モジュール名	string	プログラム内でのモジュールを一意にする名称 特殊な util のみ名称を保有
	VendorVersion	ベンダバージョン	string	下限ベンダバージョン
	UpperVendorVersion	上限ベンダバージョン	string	上限ベンダバージョン
ReturnValue		処理結果	long	メソッド実行結果

GetOperatingAuthority メソッド

以下に GetOperatingAuthority メソッドを使用するサンプルソースを示します。

- [更新適用重要度設定のサンプルソース 1](#)
- [更新適用重要度設定のサンプルソース 2](#)

GetOperatingAuthority メソッドの引数

以下に GetOperatingAuthority メソッドの引数を表記します。

引数名	項目	型	説明	引数の設定要否
AppKey	アプリケーションキー	string	操作権限を取得するアプリケーション名	必須

GetOperatingAuthority メソッドの戻り値

以下に GetOperatingAuthority メソッドの戻り値を表記します。

戻り値名	項目	型	説明
ReturnValue	処理結果	long	メソッド実行結果

ReleaseOperatingAuthority メソッド

以下に ReleaseOperatingAuthority メソッドを使用するサンプルソースを示します。

- [更新適用重要度設定のサンプルソース 1](#)
- [更新適用重要度設定のサンプルソース 2](#)

ReleaseOperatingAuthority メソッドの引数

以下に ReleaseOperatingAuthority メソッドの引数を表記します。

引数名	項目	型	説明	引数の設定要否
AppKey	アプリケーションキー	string	操作権限を取得するアプリケーション名	必須

ReleaseOperatingAuthority メソッドの戻り値

以下に ReleaseOperatingAuthority メソッドの戻り値を表記します。

戻り値名	項目	型	説明
ReturnValue	処理結果	long	メソッド実行結果

SetUpdatePriority メソッド

以下に SetUpdatePriority メソッドを使用するサンプルソースを示します。

- [更新適用重要度設定のサンプルソース 1](#)
- [更新適用重要度設定のサンプルソース 2](#)

SetUpdatePriority メソッド(デバイス用 CIM Provider)の引数

以下に SetUpdatePriority メソッド(デバイス用 CIM Provider)の引数を表記します。

引数名	項目	型	説明	引数の設定要否
AppKey	アプリケーションキー	string	操作権限を取得するアプリケーション名	必須
UpdatePriority	重要度	int	適用重要度 1~5 の間の値、9: 適用しない	必須
DeviceID	デバイス ID	string	重要度を設定するデバイスのデバイス ID	必須
ModuleKind	モジュール種別	string	重要度を設定するデバイスのモジュール種別 "firm": フームウェア、 "driver": ドライバ、 "util": ユーティリティ 適用重要度設定はモジュール種別単位でなくデバイス単位で管理するため、設定時にはデバイスで管理している全モジュール種別に対して重要度設定を行う	必須

SetUpdatePriority メソッド(ツール用 CIM Provider)の引数

以下に SetUpdatePriority メソッド(ツール用 CIM Provider)の引数を表記します。

引数名	項目	型	説明	引数の設定要否
AppKey	アプリケーションキー	string	操作権限を取得するアプリケーション名	必須
UpdatePriority	重要度	int	適用重要度 1~5 の間の値、9: 適用しない	必須

SetUpdatePriority メソッドの戻り値

以下に SetUpdatePriority メソッドの戻り値を表記します。

戻り値名	項目	型	説明
ReturnValue	処理結果	long	メソッド実行結果

GetUpdatePlans メソッド

以下に GetUpdatePlans メソッドを使用するサンプルソースを示します。

- [アップデート予定リスト取得のサンプルソース](#)

GetUpdatePlans メソッド(デバイス用 CIM Provider)の引数

以下に GetUpdatePlans メソッド(デバイス用 CIM Provider)の引数を表記します。

引数名	項目	型	説明	引数の設定要否
Rollback	ロールバック指定	bool	アップデートかロールバックかを判断するフラグ True: ロールバック、 False: アップデート	必須

GetUpdatePlans メソッド(ツール用 CIM Provider)の引数

GetUpdatePlans メソッド(ツール用 CIM Provider)に引数はありません。

GetUpdatePlans メソッド(デバイス用 CIM Provider)の戻り値

以下に GetUpdatePlans メソッド(デバイス用 CIM Provider)の戻り値を表記します。

戻り値名	項目	型	説明
UpdatePlansList	アップデート予定リスト	list<Hitachi_UpdatePlans>	デバイス群の各モジュールのアップデート予定リスト
ProviderName	プロバイダ名	string	プロバイダ名
VendorName	ベンダ名	string	ベンダ名
UniqueName	ユニーク名	string	ユニーク名
DeviceId	デバイス ID	string	デバイス ID
ModuleKind	モジュール種別	string	モジュール種別 "firm": フームウェア、"driver": ドライバ、 "util": ユーティリティ
UpdatePriority	更新適用重要度	int	更新適用重要度 1~5 の間の値
CurrentVendorVersion	稼動バージョン	string	稼動バージョン
NextVendorVersion	適用バージョン	string	更新バージョン ロールバック時は旧稼動バージョン
DisplayName	表示名称	string	適用バージョンの表示名称
UpdateJudgment	更新可否	int	更新可否 0: 更新非対称(更新適用重要度が"適用しない")、1: 更新ありアップデート可能、2: 更新ありアップデート条件 NG、3: 更新あり依存関係 NG、4: 更新なし(更新モジュールなし) 更新あり依存関係 NG はプロバイダ内と他プロバイダのチェック結果両方を示す
DependencyName	依存先 NG 名	string	プロバイダ名+ベンダ名+ユニーク名+モジュール種別をスペースで連結した文字列 更新可否項目が「更新あり依存関係 NG」を示したときのみ設定する
ReturnValue	処理結果	long	メソッド実行結果

GetUpdatePlans メソッド(ツール用 CIM Provider)の戻り値

以下に GetUpdatePlans メソッド(ツール用 CIM Provider)の戻り値を表記します。

戻り値名	項目	型	説明
UpdatePlansList	アップデート予定リスト	list<Hitachi_UpdatePlans>	デバイス群の各モジュールのアップデート予定リスト
ProviderName	プロバイダ名	string	プロバイダ名
VendorName	ベンダ名	string	ベンダ名
UniqueName	ユニーク名	string	ユニーク名
DeviceId	デバイス ID	string	デバイス ID
ModuleKind	モジュール種別	string	モジュール種別 "util": ユティリティ
UpdatePriority	更新適用重要度	int	更新適用重要度 1~5 の間の値
CurrentVendorVersion	稼動バージョン	string	稼動バージョン
NextVendorVersion	適用バージョン	string	更新バージョン ロールバック時は旧稼動バージョン
DisplayName	表示名称	string	適用バージョンの表示名称
UpdateJudgment	更新可否	int	更新可否 0: 更新非対称(更新適用重要度が"適用しない")、1: 更新ありアップデート可能、2: 更新ありアップデート条件 NG、3: 更新あり依存関係 NG、4: 更新なし(更新モジュールなし) 更新あり依存関係 NG はプロバイダ内と他プロバイダのチェック結果両方を示す
DependencyName	依存先 NG 名	string	プロバイダ名+ベンダ名+ユニーク名+モジュール種別をスペースで連結した文字列 更新可否項目が「更新あり依存関係 NG」を示したときのみ設定する
ReturnValue	処理結果	long	メソッド実行結果

HITACHI_ToolServiceWDクラス

HITACHI_ToolServiceWD クラスで提供するメソッドについて説明します。

HITACHI_ToolServiceWD クラスのメソッド一覧

以下に HITACHI_ToolServiceWD クラスのメソッドの一覧を表記します。

メソッド	機能	説明
GetParameter	値取得メソッド	設定項目の名称を指定して、その設定値を取得する
SetParameter	値設定メソッド	設定項目に値を設定する(設定のみで、プログラムには反映されない)
UpdateParameters	値更新メソッド	値設定メソッドで設定した値を更新し、プログラムに反映する
ClearParameters	値削除メソッド	値設定メソッドで設定した値を削除する

GetParameter メソッド

以下に GetParameter メソッドを使用するサンプルソースを示します。

- [監視設定取得のサンプルソース](#)

GetParameter メソッドの引数

以下に GetParameter メソッドの引数を表記します。

引数名	項目	型	説明	引数の設定要否
Name	設定項目名称	string	取得する設定項目の名称 ¥0 で終わる文字列 "ErrorWatching": エラー監視設定、 "TimeoutAction": タイムアウト処理、 "TimeoutInterval": タイムアウトインターバル	必須

GetParameter メソッドの戻り値

以下に GetParameter メソッドの戻り値を表記します。

戻り値名	項目	型	説明
Value	設定値	string	設定値 ¥0 で終わる文字列 TRUE: エラー監視を行う、FALSE: エラー監視を行わない (ErrorWatching) NoAction、HardReset、PowerDown、PowerSycle、NMI、SMI (TimeoutAction) 10~3600 までの 10 進数の整数文字列(TimeoutInterval)

戻り値名	項目	型	説明
Type	型	long	設定値の型 3:論理値(ErrorWatching)、5:列挙値(TimeoutAction)、 1:整数値(TimeoutInterval)
ReturnValue	処理結果	long	メソッド実行結果

SetParameter メソッド

以下に SetParameter メソッドを使用するサンプルソースを示します。

- [監視設定更新のサンプルソース](#)

SetParameter メソッドの引数

以下に SetParameter メソッドの引数を表記します。

引数名	項目	型	説明	引数の設定要否
Name	設定項目名称	string	取得する設定項目の名称 ¥0 で終わる文字列 "ErrorWatching": エラー監視設定、 "TimeoutAction": タイムアウト処理、 "TimeoutInterval": タイムアウトインターバル	必須
Value	設定値	string	設定値 ¥0 で終わる文字列 TRUE: エラー監視を行う、 FALSE: エラー監視を行わない(ErrorWatching) NoAction、HardReset、PowerDown、 PowerSycle、NMI、SMI(TimeoutAction) 10~3600までの10進数の整数文字列 (TimeoutInterval)	必須

SetParameter メソッドの戻り値

以下に SetParameter メソッドの戻り値を表記します。

戻り値名	項目	型	説明
ReturnValue	処理結果	long	メソッド実行結果

UpdateParameters メソッド

以下に UpdateParameters メソッドを使用するサンプルソースを示します。

- [監視設定更新のサンプルソース](#)

UpdateParameters メソッドの引数

UpdateParameters メソッドに引数はありません。

UpdateParameters メソッドの戻り値

以下に UpdateParameters メソッドの戻り値を表記します。

戻り値名	項目	型	説明
ReturnValue	処理結果	long	メソッド実行結果

ClearParameters メソッド

以下に ClearParameters メソッドを使用するサンプルソースを示します。

- [監視設定更新のサンプルソース](#)

ClearParameters メソッドの引数

ClearParameters メソッドに引数はありません。

ClearParameters メソッドの戻り値

以下に ClearParameters メソッドの戻り値を表記します。

戻り値名	項目	型	説明
ReturnValue	処理結果	long	メソッド実行結果

各クラスのメソッド共通実行結果

各クラスのメソッド共通の実行結果(ReturnValue)について説明します。

メソッドの実行結果一覧

以下にメソッドの実行結果の一覧を表記します。

メソッド実行結果(16進数)	エラー要因	ユーザ依頼事項
0	なし(正常終了)	対処不要です
11	デバイス用 CIM Provider の操作権限を取得せずに処理を実行しようとした	更新適用重要度設定処理の前に操作権限取得処理を更新適用重要度設定処理の後に操作権限解放処理を実行しているか確認 操作権限取得・更新適用重要度設定・操作権限解放処理時に設定したアプリケーションキーがすべて一致しているか確認
12	デバイス用 CIM Provider の操作権限取得に失敗した	Server Navigator の他の処理が実行中か確認し、実行中の場合は終了を待つ 更新適用重要度設定処理の前に操作権限取得処理を更新適用重要度設定処理の後に操作権限解放処理を実行しているか確認 操作権限取得・更新適用重要度設定・操作権限解放処理時に設定したアプリケーションキーがすべて一致しているか確認
1001	Update Manager とのソケット通信に失敗した	Server Navigator の他の処理が実行中か確認し、実行中の場合は終了を待つ
1101		Update Manager のサービスとプロセスが正常に開始されているか確認
1002	Update Manager とのソケット通信が競合した	Update Manager のサービスとプロセスが正常に開始されているか確認
1105	実行したメソッド(ダウンロード指示、自動ダウンロード設定、自動アップデート設定)のパラメータに禁止文字「;(半角セミコロン)」が含まれている	設定したパラメータから「;(半角セミコロン)」を消す
1106	実行した処理(ダウンロード指示、自動ダウンロード設定の更新)に失敗した	Server Navigator の他の処理が実行中か確認し、実行中の場合は終了を待つ
1107	自動アップデート設定の更新に失敗した	Update Manager のサービスとプロセスが正常に開始されているか確認
1108	ダウンロードに失敗した	外部媒体のパスやモジュールの格納の仕方に誤りが無いか確認 自動ダウンロード設定の web 接続情報に誤りが無いか確認 Server Navigator の他の処理が実行中か確認し、実行中の場合は終了を待つ Update Manager のサービスとプロセスが正常に開始されているか確認
1109	アップデートに失敗した	Server Navigator の他の処理が実行中か確認し、実行中の場合は終了を待つ
1110	ロールバックに失敗した	Update Manager のサービスとプロセスが正常に開始されているか確認

メソッド実行結果(16進数)	エラー要因	ユーザ依頼事項
1112	システムリブートに失敗した	Server Navigator の他の処理が実行中か確認し、実行中の場合は終了を待つ Update Manager のサービスとプロセスが正常に開始されているか確認 Server Navigator の他の処理が実行中で無いことを確認し、BMC デバイスのアップデートがある場合は手動シャットダウンをない場合は手動リブートを行う
1113	実行した処理(ダウンロード指示、自動ダウンロード設定の更新、自動アップデート設定の更新)に必要なパラメータが不足・誤りがある	設定したパラメータに不足・誤りがないか確認
1114	Server Navigator の他の処理と競合した	Server Navigator の他の処理が実行中か確認し、実行中の場合は終了を待つ 操作権限取得時のアプリケーションキー指定誤りが無いか確認
1115	自動ダウンロード設定の更新の場合、自動ダウンロード設定日時が 1 時間以内 自動アップデート設定の更新の場合、自動アップデート設定日時が、1 時間以内	自動ダウンロード設定の更新の場合、自動ダウンロード設定日時を 1 時間より後に変更する。 自動アップデート設定の更新の場合、自動アップデート設定日時を 1 時間より後に変更する
80000002	不明なエラー	対処方法はありません。繰り返し発生する場合は「Hitachi Server Navigator ユーザーズガイド アップデート機能」-「トラブルシューティング」に記載されている「障害解析用ログファイル」を採取し、「日立サポート 360」へお問い合わせください。
80000004	値取得メソッドの場合、パラメータ不正 (Name の指定値不正) 値設定メソッドの場合、パラメータ不正 (Name の指定値不正、Value の指定値不正、Value の有効範囲エラー) 更新適用重要度設定メソッドの場合、更新適用重要度が 1~5 の間の値でなく、9(適用しない)でもない	値取得メソッドの場合、設定項目名称を確認する 値設定メソッドの場合、設定項目名称、設定値を確認する 更新適用重要度設定メソッドの場合、更新適用重要度を 1~5 の範囲あるいは 9 に修正する

Server Navigator CLIで使用するクエリ一覧

OS 標準の WMI プロバイダの ExecQuery メソッドを使用して、OS 標準の WMI クラスよりシステム情報を取得します。また、Server Navigator のデバイス用 CIM Provider クラスの取得に使用します。

Server Navigator の GUI 操作を CLI に置き換えるために使用するクエリの一覧について説明します。

クエリ一覧

以下に Server Navigator CLI で使用するクエリの一覧を表記します。

クエリ名称	クラス	概要
PCIデバイス情報取得クエリ	Win32_PnPEntity	デバイス ID に'PCI'を含み、かつ Win32 構成マネージャーのエラーが発生していない、デバイスの現在の状態が正常であるデバイスを抽出
Baseboard詳細説明情報・製造元情報取得クエリ	Win32_ComputerSystem	詳細説明情報と製造元情報を取得する
BaseboardデバイスID(UUID)取得クエリ	Win32_ComputerSystemProduct	UUID を取得する
デバイス用CIM Providerクラス取得クエリ	-	デバイス用 CIM Provider 共通クラスをスーパークラスとするクラスを抽出
ツール用CIM Providerクラス取得クエリ	-	ツール用 CIM Provider 共通クラスをスーパークラスとするクラスを抽出

PCI デバイス情報取得クエリ

以下に PCI デバイス情報取得クエリを使用するサンプルソースを示します。

- [デバイスリスト表示のサンプルソース](#)
- [デバイス情報表示のサンプルソース](#)
- [アップデート情報表示のサンプルソース](#)
- [troubleshoot表示のサンプルソース](#)
- [ReleaseNote表示のサンプルソース](#)
- [アップデート進捗率表示のサンプルソース](#)
- [更新適用重要度表示のサンプルソース](#)
- [アップデート予定リスト取得のサンプルソース](#)
- [連携ツール表示のサンプルソース](#)

PCI デバイス情報取得クエリの内容と取得項目

以下に PCI デバイス情報取得クエリを表記します。

クラス	クエリ	概要
Win32_PnPEntity	SELECT Caption,DeviceID FROM Win32_PnPEntity WHERE DeviceID LIKE 'PCI%' AND ConfigManagerErrorCode = '0' AND Status = 'OK'	デバイス ID に'PCI'を含み、かつ Win32 構成マネージャーのエラーが発生していない、デバイスの現在の状態が正常であるデバイスを抽出
	SELECT Caption,Description,Manufacturer,DeviceID FROM Win32_PnPEntity WHERE DeviceID LIKE 'PCI%' AND ConfigManagerErrorCode = '0' AND Status = 'OK'	

以下に PCI デバイス情報取得クエリの取得項目を表記します。

取得項目名	項目	型	説明
Caption	デバイス名称(詳細)	string	デバイス名見出し
Description	詳細説明情報	string	デバイスの説明
Manufacturer	製造元情報	string	デバイスの製造メーカー(ベンダ)名
DeviceID	デバイス ID	string	4 衔のデバイス ID やベンダ ID、OS が決定するユニーク部を持つデバイス ID

Baseboard 詳細説明情報・製造元情報取得クエリ

以下に Baseboard 詳細説明情報・製造元情報取得クエリを使用するサンプルソースを示します。

● [デバイス情報表示のサンプルソース](#)

Baseboard 詳細説明情報・製造元情報取得クエリの内容と取得項目

以下に Baseboard 詳細説明情報・製造元情報取得クエリを表記します。

クラス	クエリ	概要
Win32_ComputerSystem	SELECT Model,Manufacturer FROM Win32_ComputerSystem	詳細説明情報と製造元情報を取得する

以下に Baseboard 詳細説明情報・製造元情報取得クエリ取得項目を表記します。

取得項目名	項目	型	説明
Model	詳細説明情報	string	デバイスの型情報
Manufacturer	製造元情報	string	デバイスの製造メーカー(ベンダ)名

Baseboard デバイス ID(UUID)取得クエリ

以下に Baseboard デバイス ID(UUID)取得クエリを使用するサンプルソースを示します。

- [デバイス情報表示のサンプルソース](#)

Baseboard デバイス ID(UUID)取得クエリの内容と取得項目

以下に Baseboard デバイス ID(UUID)取得クエリを表記します。

クラス	クエリ	概要
Win32_ComputerSystemProduct	SELECT UUID FROM Win32_ComputerSystemProduct	UUID を取得する

以下に Baseboard デバイス ID(UUID)取得クエリの取得項目を表記します。

取得項目名	項目	型	説明
UUID	デバイス ID	string	デバイスのユニーク ID

デバイス用 CIM Provider クラス取得クエリ

以下にデバイス用 CIM Provider クラス取得クエリを使用するサンプルソースを示します。

- [デバイスリスト表示のサンプルソース](#)
- [デバイス情報表示のサンプルソース](#)
- [アップデート情報表示のサンプルソース](#)
- [全デバイス用CIM Provider取得のサンプルソース](#)
- [更新適用重要度表示のサンプルソース](#)
- [更新適用重要度設定のサンプルソース 1](#)
- [アップデート予定リスト取得のサンプルソース](#)

デバイス用 CIM Provider クラス取得クエリの内容と取得項目

以下にデバイス用 CIM Provider クラス取得クエリを表記します。

クラス	クエリ	概要
-	Select * From meta_class Where __SuperClass = 'HITACHI_DevService'	デバイス用 CIM Provider 共通クラスをスーパークラスとするクラスを抽出

以下にデバイス用 CIM Provider クラス取得クエリの取得項目を表記します。

取得項目名	説明
デバイス用 CIM Provider のクラス	Server Navigator でサポートするデバイス用 CIM Provider クラス

ツール用 CIM Provider クラス取得クエリ

以下にツール用 CIM Provider クラス取得クエリを使用するサンプルソースを示します。

- [デバイスリスト表示のサンプルソース](#)
- [アップデート情報表示のサンプルソース](#)
- [全ツール用CIM Provider取得のサンプルソース](#)
- [更新適用重要度表示のサンプルソース](#)
- [更新適用重要度設定のサンプルソース 1](#)
- [アップデート予定リスト取得のサンプルソース](#)

ツール用 CIM Provider クラス取得クエリの内容と取得項目

以下にツール用 CIM Provider クラス取得クエリを表記します。

クラス	クエリ	概要
-	Select * From meta_class Where __SuperClass = 'HITACHI_ToolService'	ツール用 CIM Provider 共通クラスをスーパークラスとするクラスを抽出

以下にツール用 CIM Provider クラス取得クエリの取得項目を表記します。

取得項目名	説明
ツール用 CIM Provider のクラス	Server Navigator でサポートするツール用 CIM Provider クラス

Server NavigatorのCLIサンプル一覧

以下に、Server Navigator の CLI サンプルについて説明します。



Server Navigator の動作には UpdateManagerService サービスが開始状態になっている必要がありますが、本付録で提供する Server Navigator の CLI サンプルではサービスの状態を検査しておりません。

サービス停止時に、情報表示を行う CLI サンプルを実行するとエラーとはなりませんが、サービス停止前の情報採取して表示します。

特に、アップデート支持/ロールバック指示でシステムリブートが実行された後は、注意が必要です。

Server Navigator の GUI 操作の個々の操作を CLI に置き換えたサンプルについて説明します。

- [デバイスリスト表示\(CLI\)](#)
- [デバイス情報表示\(CLI\)](#)
- [アップデート情報表示\(CLI\)](#)
- [デバイス情報表示\(CLI\)](#)
- [troubleshoot表示\(CLI\)](#)
- [Release Note表示\(CLI\)](#)
- [ダウンロード指示\(CLI\)](#) *1
- [アップデート指示\(CLI\)](#)
- [ロールバック指示\(CLI\)](#)
- [全デバイス用CIMProvider取得\(CLI\)](#)
- [全ツール用CIMProvider取得\(CLI\)](#)
- [アップデート進捗率表示\(CLI\)](#)
- [自動ダウンロード設定取得\(CLI\)](#)
- [自動ダウンロード設定更新\(CLI\)](#) *1
- [自動アップデート設定取得\(CLI\)](#)
- [自動アップデート設定更新\(CLI\)](#) *1
- [トレースログ取得\(CLI\)](#)
- [更新適用重要度表示\(CLI\)](#)
- [更新適用重要度設定\(CLI\)](#) *1
- [アップデート予定リスト取得\(CLI\)](#)

- [アップデート履歴取得\(CLI\)](#)
- [監視設定取得\(CLI\)](#)
- [監視設定更新\(CLI\)](#) *1
- [連携ツール表示\(CLI\)](#)

*1 設定項目がある操作のため、サンプルソースはカスタマイズしないと使用できません。



各サンプルのサンプルソースは VBScript のソースです。拡張子".vbs"で保存して、下記コマンドで実行してください。

引数は第 1 引数のみ有効で、ホスト名または IP アドレスを与えることにより、制御するホストを指定できます。引数を省略または " ." とした場合はローカルホストを指定となります。

XXX.vbs は例です。実際のファイル名に置き換えてください。

```
"cscript.exe //NoLogo XXX.vbs [<ホスト名|IP>]"
```



サンプルソースの左枠外の(1)(2)や 1. 2. 3. は、サンプルソースの処理内容詳細の説明箇所を示します。

また、「!」はその行と次の行が一つの行であることを示します。" "(空白文字)を挟んで連結して一つの行にしてください。

デバイスリスト表示(CLI)

概要

GUI では、以下の処理を実行します。

1. システムデバイスリスト表示

システムデバイス名(固定文言 : System)をツリーの頂点ノードに表示し、システムデバイスとして BIOS/EFI デバイスと BMC デバイスのデバイス名称をツリー表示します。

システムデバイスリストで表示するデバイス名は、固定文言(BIOS/EFI デバイス : BIOS/EFI、BMC デバイス : BMC)です。

2. デバイスリスト表示

WMI のクエリを実行して、全デバイス用 CIM Provider を取得します。

BIOS/EFI デバイスと BMC デバイス以外のデバイス用 CIM Provider のプロバイダ管理情報を取得して、接続デバイス情報がある場合、接続デバイス情報を取得し、また WMI のクエリを実行して PCI デバイス情報を取得します。

デバイス用 CIM Provider のプロバイダ名をツリーの頂点ノードに表示し、接続デバイス情報のデバイス ID と取得した PCI デバイス情報のデバイス ID が一致する情報を検索し、PCI デバイス情報のデバイス名称(詳細)をツリー表示します。

3. ツールリスト表示

WMI のクエリを実行して、全ツール用 CIM Provider を取得します。

ツール用 CIM Provider のプロバイダ管理情報を取得して、接続ツール情報がある場合、接続ツール情報を取得します。

ツールリスト名(固定文言: Tool)をツリーの頂点ノードに表示し、ツール用 CIM Provider のプロバイダ名をツリー表示します。WMI のクエリを実行して、全デバイス用 CIM Provider を取得します。

4. デバイス選択状態およびアップデート情報有無アイコン表示

デバイス選択画面による選択状態とアップデート情報の有無によって表示アイコンを決定し、ツリーノード名、デバイス名の頭に表示します。

なおツリーノード名の頭に表示するアイコンは、デバイス名に表示するアイコンのうち、選択アイコン(グリーン)、選択アイコン(グレー)、未選択アイコン(ハイフン)の優先順位で最も優先順位が高いアイコンを表示します。

デバイス選択状態/アップデート情報有無	情報あり	情報なし
選択	選択アイコン(グリーン)	選択アイコン(グレー)
未選択	未選択アイコン(ハイフン)	未選択アイコン(ハイフン)

サンプルソースの処理内容詳細

サンプルソースでは、デバイス用 CIM Provider 及びツール用 CIM Provider をツリー表示します。

また、表示アイコンを選択アイコン(グリーン) : i、選択アイコン(グレー) : +、未選択アイコン(ハイフン) : _ の半角英字・記号でそれぞれ表示します。

(1) システムデバイスリスト表示

1. プロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
2. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
メソッド実行結果が異常の場合、メソッド実行結果を表示して次のデバイス用 CIM Provider のプロバイダ管理情報を取得する
3. 接続デバイス情報のモジュール情報から更新適用重要度を取得し、デバイス選択状態を判定する
4. 接続デバイス情報のモジュール情報からリポジトリ管理モジュール情報を取得し、適用種別からアップデート情報の有無を判定する
5. デバイス選択状態とアップデート情報の有無から表示アイコンを決定し、デバイス名の頭に記号を付加する
6. システムデバイス名(固定文言: System)をツリーの頂点ノードに表示し、BIOS/EFI デバイスと BMC デバイスをシステムデバイスとして、デバイス名称を固定文言 (BIOS/EFI デバイス : BIOS/EFI、BMC デバイス : BMC) で固定ツリー表示する
7. メソッド実行結果を表示する

(2) デバイスリスト表示

1. WMI のクエリを実行して全デバイス用 CIM Provider クラスを取得する
2. BIOS/EFI と BMC 以外の全デバイス用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
3. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
メソッド実行結果が異常の場合、メソッド実行結果を表示して次のデバイス用 CIM Provider のプロバイダ管理情報を取得する
4. WMI のクエリを実行して、PCI デバイス情報のデバイス名称(詳細)とデバイス ID を取得する
5. 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合、接続デバイス情報のモジュール情報から更新適用重要度を取得し、デバイス選択状態を判定する
6. 接続デバイス情報のモジュール情報からリポジトリ管理モジュール情報を取得し、適用種別からアップデート情報の有無を判定する
7. デバイス選択状態とアップデート情報の有無から表示アイコンを決定し、PCI デバイス情報のデバイス名称(詳細)の頭に記号を付加する

8. プロバイダ名をツリーの頂点ノードに表示し、PCI デバイス情報のデバイス名称(詳細)をツリー表示する
 9. メソッド実行結果を表示する
- (3) ツールリスト表示
1. WMI のクエリを実行して全ツール用 CIM Provider クラスを取得する
 2. 全ツール用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
 3. メソッド実行結果が正常かつプロバイダ管理情報の接続ツール情報が存在する場合、接続ツール情報を取得する
メソッド実行結果が異常の場合、メソッド実行結果を表示して次のツール用 CIM Provider のプロバイダ管理情報を取得する
 4. 接続ツール情報のモジュール情報から更新適用重要度を取得し、デバイス選択状態を判定する
 5. 接続ツール情報のモジュール情報からリポジトリ管理モジュール情報を取得し、適用種別からアップデート情報の有無を判定する
 6. デバイス選択状態とアップデート情報の有無から表示アイコンを決定し、プロバイダ名の頭に記号を付加する
 7. ツールリスト名(固定文言: Tool)をツリーの頂点ノードに表示し、プロバイダ名をツリー表示する
 8. メソッド実行結果を表示する

デバイスリスト表示のサンプルソース

```
' デバイスリスト表示: prtDevList.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim objClassSet, objClass
Dim objPnPSet, objPnP
Dim objDevToolInfo, objModuleInfo, objReplInfo
Dim dicDeviceToolNameList, dicDeviceToolName
Dim strTreeIcon, strDevIcon
Dim ProviderList(1), strProvider
Dim strComputer
Const BiosProvider = "HITACHI_DevServiceBios"
Const BMCProvider = "HITACHI_DevServiceBMC"
Const conTarget = "i"
Const conTargetNoModule = "+"
Const conNonTarget = "_"

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

(1)
' システムデバイスリスト表示
' システムデバイスプロバイダリストを作成
ProviderList(0) = BiosProvider
ProviderList(1) = BMCProvider
' デバイス名・メソッド実行結果保持リストを作成
set dicDeviceToolNameList = CreateObject("Scripting.Dictionary")
' アイコンを未選択（ハイフン）に設定
strTreeIcon = conNonTarget
' デバイス名・メソッド実行結果保持リストを初期化
dicDeviceToolNameList.RemoveAll
' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts:" & strComputer & "\root\cimv2")

For Each strProvider In ProviderList
    ' アイコンを未選択（ハイフン）に設定
    strDevIcon = conNonTarget
    ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod(strProvider, "GetProviderManageInfo")
    ' メソッド実行結果が正常の場合
    If objOutParams.ReturnValue = &H0 Then
        ' プロバイダ管理情報の接続デバイス情報が存在する場合
        If IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
            ' 接続デバイス情報を取得
            For Each objDevToolInfo In objOutParams.DevProviderManageInfo.DeviceInfo
                ' デバイスアイコン決定
                DecideDeviceIcon()
                ' デバイスアイコンの優先順位が高い場合は、ツリーアイコンを変更
                If strTreeIcon = conNonTarget And (strDevIcon = conTargetNoModule Or strDevIcon = conTarget) Then
                    Then
                        strTreeIcon = strDevIcon
                    ElseIf strTreeIcon = conTargetNoModule And strDevIcon = conTarget Then
                        strTreeIcon = strDevIcon
                    End If
                Next
                ' デバイスアイコンを付加したデバイス名とメソッド実行結果をリストに保持
                ' BIOS/EFI 用 CIM Provider の場合
                If strProvider = BiosProvider Then
                    dicDeviceToolNameList.Add strProvider, strDevIcon & " BIOS/EFI"
                ' BMC 用 CIM Provider の場合
                End If
            Next
        End If
    End If
End If
End If
```

```

        Elseif strProvider = BMCProvider Then
            dicDeviceToolNameList.Add strProvider, strDevIcon & "BMC"
        End If
        dicDeviceToolNameList.Add strProvider & "ReturnValue", "ReturnValue GetProviderManageInfo (" &
        Replace(strProvider, "HITACHI_DevService", "") & "): " & Hex(objOutParams.ReturnValue)
        End If
        ' メソッド実行結果が異常の場合
    Else
        ' デバイス名とメソッド実行結果をリストに保持
        ' BIOS/EFI 用 CIM Provider の場合
        If strProvider = BiosProvider Then
            dicDeviceToolNameList.Add strProvider, "BIOS/EFI"
        ' BMC 用 CIM Provider の場合
        Elseif strProvider = BMCProvider Then
            dicDeviceToolNameList.Add strProvider, "BMC"
        End If
        dicDeviceToolNameList.Add strProvider & "ReturnValue", "ReturnValue GetProviderManageInfo (" &
        Replace(strProvider, "HITACHI_DevService", "") & "): " & Hex(objOutParams.ReturnValue)
        End If
    Next
    ' システムデバイスリストをツリー表示
    Wscript.echo strTreeIcon & "System"
    For Each dicDeviceToolName In dicDeviceToolNameList
        Wscript.echo " " & dicDeviceToolNameList.Item(dicDeviceToolName)
    Next

(2)
1.     ' デバイスリスト表示
1.     ' 全デバイス用 CIM Provider クラスを取得
1.     Set objClassSet = objWMIService.ExecQuery("Select * From meta_class Where __SuperClass = 'HITACHI_DevService'")

2.     For Each objClass In objClassSet
3.         ' アイコンを未選択 (ハイフン) に設定
3.         strTreeIcon = conNonTarget
3.         strDevIcon = conNonTarget
4.         ' デバイス名・メソッド実行結果保持リストを初期化
4.         dicDeviceToolNameList.RemoveAll
5.         ' BIOS/EFI 用 CIM Provider でも BMC 用 CIM Provider でもない場合
5.         If objClass.Path_.Class <> BiosProvider And objClass.Path_.Class <> BMCProvider Then
6.             ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
6.             Set objOutParams = objWMIService.ExecMethod(objClass.Path_.Class, "GetProviderManageInfo")
7.             ' メソッド実行結果が正常の場合
7.             If objOutParams.ReturnValue = &H0 Then
8.                 ' プロバイダ管理情報の接続デバイス情報が存在する場合
8.                 If IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
9.                     ' OS から PCI デバイス情報を取得
9.                     Set objPnPSet = objWMIService.ExecQuery(
10.                         ("SELECT Caption, DeviceID FROM Win32_PnPEntity WHERE DeviceID LIKE 'PCI%' AND
10. ConfigManagerErrorCode = '0' AND Status = 'OK'"))
11.                     ' 接続デバイス情報と合致する PCI デバイス情報を検索
11.                     For Each objDevToolInfo In objOutParams.DevProviderManageInfo.DeviceInfo
12.                         For Each objPnP In objPnPSet
13.                             ' 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合
13.                             If objDevToolInfo.DeviceID = objPnP.DeviceID Then
14.                                 ' デバイスアイコン決定
14.                                 DecideDeviceIcon()
15.                                 ' デバイスアイコンの優先順位が高い場合は、ツリーアイコンを変更
15.                                 If strTreeIcon = conNonTarget And (strDevIcon = conTargetNoModule Or strDevIcon =
16. conTarget) Then
16.                                     strTreeIcon = strDevIcon
17.                                 Elseif strTreeIcon = conTargetNoModule And strDevIcon = conTarget Then
17.                                     strTreeIcon = strDevIcon
18.                                 End If
19.                                 ' デバイスアイコンを付加したデバイス名をリストに保持
19.                                 dicDeviceToolNameList.Add objPnP.DeviceID, strDevIcon & objPnP.Caption
20.                                 Exit For
20.                             End If

```

```

        Next
8.      Next
        ' デバイスリストをツリー表示
        Wscript.echo strTreeIcon & objOutParams.DevProviderManageInfo.ProviderName
        For Each dicDeviceToolName In dicDeviceToolNameList
            Wscript.echo " " & dicDeviceToolNameList.Item(dicDeviceToolName)
        Next
        ' メソッド実行結果を表示
        Wscript.echo " ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
        "HITACHI_DevService", "") & "): " & Hex(objOutParams.ReturnValue)
        End If
        ' メソッド実行結果が異常の場合
        Else
            ' メソッド実行結果を表示
            Wscript.echo "ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
            "HITACHI_DevService", "") & "): " & Hex(objOutParams.ReturnValue)
        End If
        End If
    Next

(3)
        ' ツールリスト表示
        ' アイコンを未選択（ハイフン）に設定
        strTreeIcon = conNonTarget
        ' デバイス名・メソッド実行結果保持リストを初期化
        dicDeviceToolNameList.RemoveAll
1.      ' 全ツール用 CIM Provider クラスを取得
        Set objClassSet = objWMIService.ExecQuery("Select \* From meta\_class Where \_SuperClass='HITACHI\_ToolService'")
        For Each objClass In objClassSet
            ' アイコンを未選択（ハイフン）に設定
            strDevIcon = conNonTarget
            ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
            Set objOutParams = objWMIService.ExecMethod(objClass.Path_.Class, "GetProviderManageInfo")
            ' メソッド実行結果が正常の場合
            If objOutParams.ReturnValue = &H0 Then
                ' プロバイダ管理情報の接続ツール情報が存在する場合
                If IsNull(objOutParams.ToolProviderManageInfo.ToolInfo) = False Then
                    ' 接続ツール情報を取得
                    For Each objDevToolInfo In objOutParams.ToolProviderManageInfo.ToolInfo
                        ' デバイスアイコン決定
                        DecideDeviceIcon()
                        ' デバイスアイコンの優先順位が高い場合は、ツリーアイコンを変更
                        If strTreeIcon = conNonTarget And (strDevIcon = conTargetNoModule Or strDevIcon = conTarget)
                    Then
                        strTreeIcon = strDevIcon
                    ElseIf strTreeIcon = conTargetNoModule And strDevIcon = conTarget Then
                        strTreeIcon = strDevIcon
                    End If
                    ' デバイスアイコンを付加したプロバイダ名をリストに保持
                    dicDeviceToolNameList.Add objOutParams.ToolProviderManageInfo.ProviderName, strDevIcon &
                    objOutParams.ToolProviderManageInfo.ProviderName
                End If
            End If
            ' メソッド実行結果をリストに保持
            dicDeviceToolNameList.Add objClass.Path_.Class & "ReturnValue", "ReturnValue
            GetProviderManageInfo (" & Replace(objClass.Path_.Class, "HITACHI_ToolService", "") & "): " &
            Hex(objOutParams.ReturnValue)
            End If
            ' メソッド実行結果が異常の場合
            Else
                ' メソッド実行結果をリストに保持
                dicDeviceToolNameList.Add objClass.Path_.Class & "ReturnValue", "ReturnValue GetProviderManageInfo (" &
                Replace(objClass.Path_.Class, "HITACHI_ToolService", "") & "): " & Hex(objOutParams.ReturnValue)
            End If
        Next
        ' ツールリストをツリー表示
        Wscript.echo strTreeIcon & "Tool"

```

```

For Each dicDeviceToolName In dicDeviceToolNameList
    Wscript.echo " " & dicDeviceToolNameList.Item(dicDeviceToolName)
Next

Set dicDeviceToolNameList = Nothing
Set objReplInfo = Nothing
Set objModuleInfo = Nothing
Set objDevToolInfo = Nothing
Set objPnP = Nothing
Set objPnPSet = Nothing
Set objClass = Nothing
Set objClassSet = Nothing
Set objOutParams = Nothing
Set objWMIService = Nothing

' /////////////////////////////////////////////////
' デバイスアイコン決定
' /////////////////////////////////////////////////
Sub DecideDeviceIcon()

    ' デバイスアイコンを未選択（ハイフン）で初期化
    strDevIcon = conNonTarget

    ' 接続デバイス・ツール情報のモジュール情報が存在する場合
    If IsNull(objDevToolInfo.ModuleInfo) = False Then
        ' モジュール情報を取得
        For Each objModuleInfo In objDevToolInfo.ModuleInfo
            ' モジュール情報の更新適用重要度が'適用しない'の場合
            If objModuleInfo.UpdatePriority = 9 Then
                ' デバイスアイコンを未選択（ハイフン）に設定
                strDevIcon = conNonTarget
                Exit For
            ' モジュール情報の更新適用重要度が'適用しない'でない場合
            Else
                ' デバイスアイコンが選択（グリーン）でない場合
                If strDevIcon <> conTarget Then
                    ' デバイスアイコンを選択（グレー）に設定
                    strDevIcon = conTargetNoModule
                End If
                ' リポジトリ管理モジュール情報を取得
                For Each objReplInfo In objModuleInfo.RepositoryInfo
                    ' 次回適用バージョンの場合
                    If objReplInfo.ApplyKind = "update" Then
                        ' デバイスアイコンを選択（グリーン）に設定
                        strDevIcon = conTarget
                        Exit For
                    End If
                Next
            End If
        Next
    End If
    ' 接続デバイス情報のモジュール情報が存在しない場合
    Else
        ' デバイスアイコンを未選択（ハイフン）に設定
        strDevIcon = conNonTarget
    End If
End Sub

```

デバイス情報表示(CLI)

概要

GUI では、以下の処理を実行します。(ツール用 CIM Provider は、デバイス情報は全て空白で表示します)

1. システムデバイス情報表示

WMI のクエリを実行して、Baseboard の詳細説明情報、製造元情報、デバイス ID(UUID)を取得します。

取得した詳細説明情報、製造元情報、デバイス ID(UUID)を BIOS/EFI デバイスと BMC デバイスの両デバイスのデバイス情報として表示します。

2. デバイス情報表示

WMI のクエリを実行して、全デバイス用 CIM Provider を取得します。

BIOS/EFI デバイスと BMC デバイス以外のデバイス用 CIM Provider のプロバイダ管理情報を取得して、接続デバイス情報がある場合、接続デバイス情報を取得し、また WMI のクエリを実行して PCI デバイス情報を取得します。

接続デバイス情報のデバイス ID と取得した PCI デバイス情報のデバイス ID が一致する情報を検索し、PCI デバイス情報の詳細説明情報、製造元情報、接続デバイス情報のデバイス ID('DEV_'の後ろから 4 衔切り出し)、稼動ハードバージョンを表示します。

RAID 用 CIM Provider の場合、稼動ハードバージョンは表示しません。

各デバイス用 CIM Provider 每の補足情報を表示します。RAID 用 CIM Provider はコントローラ番号(接続デバイス情報の稼動ハードバージョン)を表示します。

デバイス用 CIM Provider	表示する補足情報の内容
RAID 用 CIM Provider	コントローラ番号
LAN 用 CIM Provider	MAC アドレス
FC 用 CIM Provider	WWN アドレス

サンプルソースの処理内容詳細

サンプルソースでは、デバイス用 CIM Provider のデバイス情報をデバイスリストのようにツリー表示をして一括で表示します。

ツール用 CIM Provider のデバイス情報は取得・表示を行いません。

(1) システムデバイス情報表示

1. WMI のクエリを実行し、Baseboard の詳細説明情報と製造元情報を取得する
2. WMI のクエリを実行し、Baseboard のデバイス ID(UUID)を取得する

3. BIOS/EFI デバイスをシステムデバイスとして固定ツリー表示し、その下に詳細説明情報と製造元情報、デバイス ID を表示する
4. BMC デバイスをシステムデバイスとして固定ツリー表示し、その下に詳細説明情報と製造元情報、デバイス ID を表示する

(2) デバイスリスト表示

1. WMI のクエリを実行して全デバイス用 CIM Provider クラスを取得する
2. BIOS/EFI と BMC 以外の全デバイス用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
3. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
メソッド実行結果が異常の場合、メソッド実行結果を表示して次のデバイス用 CIM Provider のプロバイダ管理情報を取得する
4. WMI のクエリを実行して、PCI デバイス情報のデバイス名称(詳細)、詳細説明情報、製造元情報、デバイス ID を取得する
5. プロバイダ名をツリーの頂点ノードに表示する
6. 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合、PCI デバイス情報のデバイス名称(詳細)をツリー表示する
7. 詳細説明情報、製造元情報を表示する
8. デバイス ID を'DEV_'の後ろから 4 衔切り出して表示する
デバイス ID の切り出しが出来ない場合は、デバイス ID を表示しない
9. 接続デバイス情報の拡張情報をキーとバリューに分割して表示する
拡張情報がキーとバリューに分割できない場合は、拡張情報を表示しない
10. メソッド実行結果を表示する

デバイス情報表示のサンプルソース

```
' デバイス情報表示: prtDevInfo.vbs
Option Explicit

Dim objWMIService
Dim objComSysSet, objComSys
Dim objComSysPSet, objComSysP
Dim objOutParams
Dim objClassSet, objClass
Dim objPnPSet, objPnP
Dim objDevInfo, objExtendedInfo
Dim intDevPos, intDevId, intExtendedPos
Dim strComputer
Const BiosProvider = "HITACHI_DevServiceBios"
Const BMCProvider = "HITACHI_DevServiceBMC"

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

(1) ' システムデバイス情報表示
' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts:" & strComputer & "\root\cimv2")

1. ' 詳細説明情報と製造元情報を取得
Set objComSysSet = objWMIService.ExecQuery(_
    ("SELECT Model, Manufacturer FROM Win32_ComputerSystem"))
2. ' デバイス ID を取得
Set objComSysPSet = objWMIService.ExecQuery(_
    ("SELECT UUID FROM Win32_ComputerSystemProduct"))

3.4. ' システムデバイス情報をツリー表示
Wscript.echo "System"
Wscript.echo " BIOS/EFI"
' 取得したデバイス情報を表示
For Each objComSys In objComSysSet
    Wscript.Echo "    Description: " & objComSys.Model
    Wscript.Echo "    Manufacturer: " & objComSys.Manufacturer
Next
For Each objComSysP In objComSysPSet
    Wscript.Echo "    Device ID: " & objComSysP.UUID
Next
Wscript.echo " BMC"
' 取得したデバイス情報を表示
For Each objComSys In objComSysSet
    Wscript.Echo "    Description: " & objComSys.Model
    Wscript.Echo "    Manufacturer: " & objComSys.Manufacturer
Next
For Each objComSysP In objComSysPSet
    Wscript.Echo "    Device ID: " & objComSysP.UUID
Next

(2) ' デバイス情報表示
1. ' 全デバイス用 CIM Provider クラスを取得
Set objClassSet = objWMIService.ExecQuery("Select * From meta_class Where __SuperClass = 'HITACHI_DevService'")

For Each objClass In objClassSet
    ' BIOS/EFI 用 CIM Provider でも BMC 用 CIM Provider でもない場合
    If objClass.Path_.Class <> BiosProvider And objClass.Path_.Class <> BMCProvider Then
        ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
        Set objOutParams = objWMIService.ExecMethod(objClass.Path_.Class, "GetProviderManageInfo")
```

```

3.      ' メソッド実行結果が正常の場合
4.      If objOutParams.ReturnValue = &H0 Then
5.          ' プロバイダ管理情報の接続デバイス情報が存在する場合
6.          If IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
7.              ' OS から PCI デバイス情報を取得
8.              Set objPnPSet = objWMIService.ExecQuery _
9.                  ("SELECT Caption,Description,Manufacturer,DeviceID FROM Win32_PnPEntity WHERE DeviceID LIKE
10.                  'PCI%' AND ConfigManagerErrorCode = '0' AND Status = 'OK'")
11.          ' プロバイダ名を表示
12.          Wscript.Echo objOutParams.DevProviderManageInfo.ProviderName
13.          ' 接続デバイス情報と合致する PCI デバイス情報を検索
14.          For Each objDeviceInfo In objOutParams.DevProviderManageInfo.DeviceInfo
15.              For Each objPnP In objPnPSet
16.                  ' 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合
17.                  If objDeviceInfo.DeviceID = objPnP.DeviceID Then
18.                      ' デバイス情報をツリー表示
19.                      ' デバイス名称を表示
20.                      Wscript.Echo "" & objPnP.Caption
21.                      ' 詳細説明情報を表示
22.                      Wscript.Echo "" Description: " & objPnP.Description
23.                      ' 製造元情報を表示
24.                      Wscript.Echo "" Manufacturer: " & objPnP.Manufacturer
25.                      ' デバイス ID の切り出し位置を取得
26.                      intDevPos = InStr(objPnP.DeviceID, "DEV_")
27.                      ' デバイス ID の切り出しが可能な場合
28.                      If intDevPos <> 0 Then
29.                          ' デバイス ID を表示 (4 桁切り出し)
30.                          Wscript.Echo " Device ID: " & Mid(objPnP.DeviceID, intDevPos + 4, 4)
31.                      End If
32.                      ' 拡張情報が存在する場合
33.                      If IsNull(objDeviceInfo.ExtendedInfo) = False Then
34.                          ' 拡張情報を取得
35.                          For Each objExtendedInfo In objDeviceInfo.ExtendedInfo
36.                              ' 拡張情報の切り出し位置を取得
37.                              intExtendedPos = InStr(objExtendedInfo, "?")
38.                              ' 拡張情報の切り出しが可能な場合
39.                              If intExtendedPos <> 0 Then
40.                                  ' 拡張情報を表示
41.                                  Wscript.Echo "" & Mid(objExtendedInfo, 1, intExtendedPos - 1) & ":" & Mid(objExtendedInfo, intExtendedPos + 1)
42.                              End If
43.                          Next
44.                      End If
45.                      Exit For
46.                  End If
47.              Next
48.          Next
49.          ' メソッド実行結果を表示
50.          Wscript.echo "ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
51.          "HITACHI_DevService", "") & "): " & Hex(objOutParams.ReturnValue)
52.      End If
53.      ' メソッド実行結果が異常の場合
54.      Else
55.          ' メソッド実行結果を表示
56.          Wscript.echo "ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
57.          "HITACHI_DevService", "") & "): " & Hex(objOutParams.ReturnValue)
58.      End If
59.  End If
60. Next

61. Set objExtendedInfo = Nothing
62. Set objDeviceInfo = Nothing
63. Set objPnP = Nothing
64. Set objPnPSet = Nothing
65. Set objClass = Nothing
66. Set objClassSet = Nothing

```

```
Set objOutParams = Nothing
Set objComSysP = Nothing
Set objComSysPSet = Nothing
Set objComSys = Nothing
Set objComSysSet = Nothing
Set objWMIService = Nothing
```

アップデート情報表示(CLI)

概要

GUI では、以下の処理を実行します。

1. デバイスアップデート情報表示(デバイス用 CIM Provider)

WMI のクエリを実行して、全デバイス用 CIM Provider を取得します。

デバイス用 CIM Provider のプロバイダ管理情報を取得して、接続デバイス情報がある場合、接続デバイス情報を取得します。

接続デバイス情報のモジュール情報がある場合、モジュール情報のアップデートステータスを取得します。

モジュール情報のリポジトリ管理モジュール情報を取得し、稼動バージョン、更新バージョン、モジュール名、重要度、リブート要否を取得します。

次回適用リポジトリ管理モジュール情報が有る場合は、稼動バージョン、更新バージョン、更新モジュール名、重要度、リブート要否を取得します。

次回適用リポジトリ管理モジュール情報が無く、稼動リポジトリ管理モジュール情報が有る場合は、稼動バージョン、稼動バージョンモジュール名を取得します。

次回適用リポジトリ管理モジュール情報が無く、稼動リポジトリ管理モジュール情報も無い場合は、稼動バージョンのみ取得します。

アップデートステータスを文言に変換して表示し、稼動バージョン、更新バージョン、モジュール名、重要度、リブート要否は、取得した項目のみ表示します。

2. アップデート情報表示(ツール用 CIM Provider)

WMI のクエリを実行して、全ツール用 CIM Provider を取得します。

ツール用 CIM Provider のプロバイダ管理情報を取得して、接続ツール情報がある場合、接続ツール情報を取得します。

接続ツール情報のモジュール情報がある場合、モジュール情報のアップデートステータスを取得します。

モジュール情報のリポジトリ管理モジュール情報を取得し、稼動バージョン、更新バージョン、モジュール名、重要度、リブート要否を取得します。

次回適用リポジトリ管理モジュール情報が有る場合は、稼動バージョン、更新バージョン、更新モジュール名、重要度、リブート要否を取得します。

次回適用リポジトリ管理モジュール情報が無く、稼動リポジトリ管理モジュール情報が有る場合は、稼動バージョン、稼動バージョンモジュール名を取得します。

次回適用リポジトリ管理モジュール情報が無く、稼動リポジトリ管理モジュール情報も無い場合は、稼動バージョンのみ取得します。

アップデートステータスを文言に変換して表示し、稼動バージョン、更新バージョン、モジュール名、重要度、リブート要否は、取得した項目のみ表示します。

サンプルソースの処理内容詳細

サンプルソースでは、デバイス用 CIM Provider 及びツール用 CIM Provider のアップデート情報をデバイスリストのようにツリー表示をして一括で表示します。

(1) システムデバイスアップデート情報表示

1. BIOS/EFI 用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
2. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
メソッド実行結果が異常あるいはプロバイダ管理情報の接続デバイス情報が存在しない場合、メソッド実行結果を表示して、BMC 用 CIM Provider のアップデート情報表示処理へ進む
3. システムデバイス名(固定文言:System)をツリーの頂点ノードに表示する
4. 接続デバイス情報を元に、アップデート情報を取得して表示する
5. メソッド実行結果を表示する
6. BMC 用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
7. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
メソッド実行結果が異常あるいはプロバイダ管理情報の接続デバイス情報が存在しない場合、メソッド実行結果を表示して、次のデバイス用 CIM Provider のプロバイダ管理情報を取得する
8. 接続デバイス情報を元に、アップデート情報を取得して表示する
9. メソッド実行結果を表示する

(2) デバイスアップデート情報表示

1. WMI のクエリを実行して全デバイス用 CIM Provider クラスを取得する
2. BIOS/EFI と BMC 以外の全デバイス用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
3. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
メソッド実行結果が異常あるいはプロバイダ管理情報の接続デバイス情報が存在しない場合、メソッド実行結果を表示して次のデバイス用 CIM Provider のプロバイダ管理情報を取得する
4. WMI のクエリを実行して、PCI デバイス情報のデバイス名称(詳細)とデバイス ID を取得する
5. プロバイダ名をツリーの頂点ノードに表示する
6. 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合、PCI デバイス情報のデバイス名称(詳細)をツリー表示する

7. 接続デバイス情報の拡張情報をキーとバリューに分割して表示する
拡張情報がキーとバリューに分割できない場合は、拡張情報を表示しない
8. 接続デバイス情報を元に、アップデート情報を取得して表示する
9. メソッド実行結果を表示する

(3) ツールアップデート情報表示

1. WMI のクエリを実行して全ツール用 CIM Provider クラスを取得する
2. 全ツール用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
3. メソッド実行結果が正常かつプロバイダ管理情報の接続ツール情報が存在する場合、接続ツール情報を取得する
メソッド実行結果が異常の場合、メソッド実行結果を表示して次のツール用 CIM Provider のプロバイダ管理情報を取得する
4. ツールリスト名(固定文言: Tool)をツリーの頂点ノードに表示する
5. 接続ツール情報を元に、アップデート情報を取得して表示する
6. メソッド実行結果を表示する

● アップデート情報表示処理

1. 接続デバイス/ツール情報のモジュール情報が存在する場合、モジュール情報を取得し、モジュール種別とアップデートステータスを表示する
アップデートステータスは、ステータス文言に変換する
2. リポジトリ管理モジュール情報を取得する
3. リポジトリ管理モジュール情報の適用種別が稼動バージョンの場合、稼動バージョンと稼動モジュールファイル名(稼動バージョンモジュールがダウンロードされているときのみ)を取得する
リポジトリ管理モジュール情報の適用種別が次回適用バージョンの場合、更新モジュールファイル名(稼動モジュール名を取得済でも、更新モジュールファイル名を優先して取得)、更新バージョン、重要度を取得し、アップデート前処理後リブート要否、アップデート後リブート要否、リストア後リブート要否のいずれかがリブート/シャットダウン要の場合、リブート要否を"True"に決定し、そうでない場合はリブート要否を"False"に決定する
4. 取得したアップデート情報を表示する

アップデート情報/ リポジトリ管理状況	管理モジュールなし	稼動バージョンモジュール管理	更新バージョンモジュール管理
アップデートステータス	表示	表示	表示
稼動バージョン	表示	表示	表示
更新バージョン	表示しない	表示しない	表示
重要度	表示しない	表示しない	表示

● アップデートステータス変換処理

1. ステータスをステータス文言に変換する

以下にステータス文言の一覧を表記します。

ステータス (16進数)	ステータス文言	ステータス
0	There is no update history.	アップデート履歴なし
10	Executing backup	バックアップ中
11	Backup completion	バックアップ完了
12	Backup failure	バックアップ失敗
20	Executing update preprocessing	アップデート前処理中
21	Waiting for reboot.(Update preprocessing)	アップデート前処理完了／リブート待ち
22	Update preprocessing completion.	アップデート前処理完了
23	Update preprocessing failure	アップデート前処理失敗
30	Executing update	アップデート中
31	Waiting for reboot.(Update)	アップデート完了／リブート待ち
32	Update completion	アップデート完了
33	Update failure	アップデート失敗
40	Executing restore	リストア中
41	Waiting for reboot.(Restore)	リストア完了／リブート待ち
42	Restore completion	リストア完了
50	Executing an application	アップデート適用判定中
51	Application success	アップデート適用成功
52	Application failure	アップデート適用失敗
上記以外	There is no update history.	アップデート履歴なし

アップデート情報表示のサンプルソース

```
' アップデート情報表示: prtUpdateInfo.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim objClassSet, objClass
Dim objPnPSet, objPnP
Dim objDevToolInfo, objModuleInfo, objReplInfo, objExtendedInfo
Dim intExtendedPos
Dim ProviderList(1), strProvider
Dim strComputer
Const BiosProvider = "HITACHI_DevServiceBios"
Const BMCProvider = "HITACHI_DevServiceBMC"

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

(1)   ' システムデバイスアップデート情報表示
      ' システムデバイスプロバイダリストを作成
ProviderList(0) = BiosProvider
ProviderList(1) = BMCProvider

      ' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\cimv2")

      ' システムデバイスツリー表示
Wscript.echo "System"
For Each strProvider In ProviderList
    ' BIOS/EFI 用 CIM Provider の場合
    If strProvider = BiosProvider Then
        ' BIOS/EFI デバイスツリー表示
        Wscript.echo " BIOS/EFI"
    ' BMC 用 CIM Provider の場合
    ElseIf strProvider = BMCProvider Then
        ' BMC デバイスツリー表示
        Wscript.echo " BMC"
    End If
    ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod(strProvider, "GetProviderManageInfo")
    ' メソッド実行結果が正常の場合
If objOutParams.ReturnValue = &H0 Then
    ' プロバイダ管理情報の接続デバイス情報が存在する場合
    If IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
        ' 接続デバイス情報を取得
        For Each objDevToolInfo In objOutParams.DevProviderManageInfo.DeviceInfo
            ' システムデバイスアップデート情報をツリー表示
            Call DisplayUpdateInfo()
        Next
        ' メソッド実行結果を表示
        Wscript.echo " ReturnValue GetProviderManageInfo (" & Replace(strProvider, "HITACHI_DevService", "") & ") : " & Hex(objOutParams.ReturnValue)
    End If
    ' メソッド実行結果が異常の場合
Else
    ' メソッド実行結果を表示
    Wscript.echo " ReturnValue GetProviderManageInfo (" & Replace(strProvider, "HITACHI_DevService", "") & ") : " & Hex(objOutParams.ReturnValue)
End If
Next
!
```

```

(2)
1.   ' デバイスアップデート情報表示
2.   ' 全デバイス用 CIM Provider クラスを取得
3.   Set objClassSet = objWMIService.ExecQuery("Select * From meta_class Where __SuperClass = 'HITACHI_DevService'")

4.   For Each objClass In objClassSet
5.     ' BIOS/EFI 用 CIM Provider でも BMC 用 CIM Provider でもない場合
6.     If objClass.Path_.Class <> BiosProvider And objClass.Path_.Class <> BMCProvider Then
7.       ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
8.       Set objOutParams = objWMIService.ExecMethod(objClass.Path_.Class, "GetProviderManageInfo")
9.       ' メソッド実行結果が正常の場合
10.      If objOutParams.ReturnValue = &H0 Then
11.        ' プロバイダ管理情報の接続デバイス情報が存在する場合
12.        If IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
13.          ' OS から PCI デバイス情報を取得
14.          Set objPnPSet = objWMIService.ExecQuery(
15.            ("SELECT Caption,DeviceID FROM Win32_PnPEntity WHERE DeviceID LIKE 'PCI%'" AND
16.             ConfigManagerErrorCode = '0' AND Status = 'OK'"))
17.          ' プロバイダ名を表示
18.          Wscript.Echo objOutParams.DevProviderManageInfo.ProviderName
19.          ' 接続デバイス情報を取得
20.          For Each objDevToolInfo In objOutParams.DevProviderManageInfo.DeviceInfo
21.            ' 接続デバイス情報と合致する PCI デバイス情報を検索
22.            For Each objPnP In objPnPSet
23.              ' 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合
24.              If objDevToolInfo.DeviceID = objPnP.DeviceID Then
25.                ' デバイス名称をツリー表示
26.                Wscript.Echo " " & objPnP.Caption
27.                ' 拡張情報が存在する場合
28.                If IsNull(objDevToolInfo.ExtendedInfo) = False Then
29.                  ' 拡張情報を取得
30.                  For Each objExtendedInfo In objDevToolInfo.ExtendedInfo
31.                    ' 拡張情報の切り出し位置を取得
32.                    intExtendedPos = InStr(objExtendedInfo, "?")
33.                    ' 拡張情報の切り出しが可能な場合
34.                    If intExtendedPos <> 0 Then
35.                      ' 拡張情報を表示
36.                      Wscript.Echo " " & Mid(objExtendedInfo, 1, intExtendedPos - 1) & ":" & Mid(objExtendedInfo, intExtendedPos + 1)
37.                    End If
38.                  Next
39.                End If
40.                ' デバイスアップデート情報をツリー表示
41.                Call DisplayUpdateInfo()
42.                Exit For
43.              End If
44.            Next
45.          Next
46.          ' メソッド実行結果を表示
47.          Wscript.echo "ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
48.            "HITACHI_DevService", "") & "):" & Hex(objOutParams.ReturnValue)
49.        End If
50.        ' メソッド実行結果が異常の場合
51.      Else
52.        ' メソッド実行結果を表示
53.        Wscript.echo "ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
54.          "HITACHI_DevService", "") & "):" & Hex(objOutParams.ReturnValue)
55.      End If
56.    End If
57.  Next

(3)
1.   ' ツールアップデート情報表示
2.   ' 全ツール用 CIM Provider クラスを取得
3.   Set objClassSet = objWMIService.ExecQuery("Select * From meta_class Where __SuperClass = 'HITACHI_ToolService'")

4.   ' ツールツリー表示

```

```

4. Wscript.echo "Tool"
For Each objClass In objClassSet
  ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
  Set objOutParams = objWMIService.ExecMethod(objClass.Path_.Class, "GetProviderManageInfo")
  ' メソッド実行結果が正常の場合
  If objOutParams.ReturnValue = &H0 Then
    ' プロバイダ管理情報の接続ツール情報が存在する場合
    If IsNull(objOutParams.ToolProviderManageInfo.ToolInfo) = False Then
      ' プロバイダ名を表示
      Wscript.Echo " " & objOutParams.ToolProviderManageInfo.ProviderName
      ' 接続ツール情報を取得
      For Each objDevToolInfo In objOutParams.ToolProviderManageInfo.ToolInfo
        ' ツールアップデート情報をツリー表示
        Call DisplayUpdateInfo()
      Next
      ' メソッド実行結果を表示
      Wscript.echo " ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
      "HITACHI_ToolService", "") & "): " & Hex(objOutParams.ReturnValue)
    End If
    ' メソッド実行結果が異常の場合
    Else
      ' メソッド実行結果を表示
      Wscript.echo " ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
      "HITACHI_ToolService", "") & "): " & Hex(objOutParams.ReturnValue)
    End If
  Next

  Set objExtendedInfo = Nothing
  Set objReplInfo = Nothing
  Set objModuleInfo = Nothing
  Set objDevToolInfo = Nothing
  Set objPnP = Nothing
  Set objPnPSet = Nothing
  Set objClass = Nothing
  Set objClassSet = Nothing
  Set objOutParams = Nothing
  Set objWMIService = Nothing

  ' /////////////////////////////////
  ' アップデート情報表示
  ' ///////////////////////////////
Sub DisplayUpdateInfo()

  Dim strCurrentVer, strNextVer, strModuleName, strDispModuleName, intPriority, strReboot
  ' 接続デバイス・ツール情報のモジュール情報が存在する場合
  If IsNull(objDevToolInfo.ModuleInfo) = False Then
    ' モジュール情報を取得
    For Each objModuleInfo In objDevToolInfo.ModuleInfo
      ' モジュール種別を表示
      Wscript.echo " [" & objModuleInfo.ModuleKind & "]"
      ' アップデートステータスを表示
      Wscript.echo " Status: " & ConvertUpdateStatus(objModuleInfo.UpdateStatus)
      ' 表示値を初期化
      strCurrentVer = Null
      strNextVer = Null
      strDispModuleName = Null
      intPriority = Null
      strReboot = Null
    Next
    ' リポジトリ管理モジュール情報を取得
    For Each objReplInfo In objModuleInfo.RepositoryInfo
      ' 稼動バージョンの場合
      If objReplInfo.ApplyKind = "current" Then
        ' 稼動バージョンを取得
        strCurrentVer = objReplInfo.VendorVersion
      End If
    Next
  End If
End Sub

```

```

        ' ファイル名を取得
        If IsNull(objRepInfo.ModuleFileName) = False Then
            For Each strModuleName In objRepInfo.ModuleFileName
                ' ファイル名を取得
                strDispModuleName = strModuleName
                Exit For
            Next
        End If
        ' 次回適用バージョンの場合
        ElseIf objRepInfo.ApplyKind = "update" Then
            ' ファイル名を取得
            If IsNull(objRepInfo.ModuleFileName) = False Then
                For Each strModuleName In objRepInfo.ModuleFileName
                    strDispModuleName = strModuleName
                    Exit For
                Next
            End If
            ' 更新バージョンを取得
            strNextVer = objRepInfo.VendorVersion
            ' 重要度を取得
            intPriority = objRepInfo.Priority
            ' リブート要否を決定
            If objRepInfo.RebootAfterPreUpdate = "message" Or _
                objRepInfo.RebootAfterPreUpdate = "force" Or _
                objRepInfo.RebootAfterPreUpdate = "shutdown" Or _
                objRepInfo.RebootAfterUpdate = "message" Or _
                objRepInfo.RebootAfterUpdate = "force" Or _
                objRepInfo.RebootAfterUpdate = "shutdown" Or _
                objRepInfo.RebootAfterRestore = "message" Or _
                objRepInfo.RebootAfterRestore = "force" Or _
                objRepInfo.RebootAfterRestore = "shutdown" Then
                strReboot = "True"
            Else
                strReboot = "False"
            End If
        End If
        Next

        ' 取得したアップデート情報を表示
        If IsNull(strCurrentVer) = False Then
            Wscript.echo "    Current Version: " & strCurrentVer
        End If
        If IsNull(strNextVer) = False Then
            Wscript.echo "    Next Version: " & strNextVer
        End If
        If IsNull(intPriority) = False Then
            Wscript.echo "    Priority: " & intPriority
        End If
        If IsNull(strReboot) = False Then
            Wscript.echo "    Reboot: " & strReboot
        End If
        If IsNull(strDispModuleName) = False Then
            Wscript.echo "    Module Name: " & strDispModuleName
        End If
        Next
    End If
End Sub

' ///////////////////////////////
' アップデートステータス変換
' ///////////////////////////////
Function ConvertUpdateStatus(Status)

    Select Case Status
    Case &H0
        ' アップデート履歴なし
    End Select
End Function

```

```

ConvertUpdateStatus = "There is no update history."
Case &H10
  ' アップデートバックアップ中
  ConvertUpdateStatus = "Executing backup"
Case &H11
  ' アップデートバックアップ完了
  ConvertUpdateStatus = "Backup completion"
Case &H12
  ' アップデートバックアップ失敗
  ConvertUpdateStatus = "Backup failure"
Case &H20
  ' アップデート前処理中
  ConvertUpdateStatus = "Executing update preprocessing"
Case &H21
  ' アップデート前処理完了／リブート待ち
  ConvertUpdateStatus = "Waiting for reboot. (Update preprocessing)"
Case &H22
  ' アップデート前処理完了
  ConvertUpdateStatus = "Update preprocessing completion."
Case &H23
  ' アップデート前処理失敗
  ConvertUpdateStatus = "Update preprocessing failure"
Case &H30
  ' アップデート中
  ConvertUpdateStatus = "Executing update"
Case &H31
  ' アップデート完了／リブート待ち
  ConvertUpdateStatus = "Waiting for reboot. (Update)"
Case &H32
  ' アップデート完了
  ConvertUpdateStatus = "Update completion"
Case &H33
  ' アップデート失敗
  ConvertUpdateStatus = "Update failure"
Case &H40
  ' アップデイトリストア中
  ConvertUpdateStatus = "Executing restore"
Case &H41
  ' アップデイトリストア完了／リブート待ち
  ConvertUpdateStatus = "Waiting for reboot. (Restore)"
Case &H42
  ' アップデイトリストア完了
  ConvertUpdateStatus = "Restore completion"
Case &H50
  ' アップデート適用判定中
  ConvertUpdateStatus = "Executing an application"
Case &H51
  ' アップデート適用成功
  ConvertUpdateStatus = "Application success"
Case &H52
  ' アップデート適用失敗
  ConvertUpdateStatus = "Application failure"
Case Else
  ' アップデート履歴なし
  ConvertUpdateStatus = "There is no update history."
End Select
End Function

```

バージョン情報表示(CLI)

概要

GUI では、以下の処理を実行します。

1. バージョン収集処理

バージョン取得 CIM Provider のバージョン収集メソッドを実行し、取得したバージョン情報を表示します。

サンプルソースの処理内容詳細

サンプルソースでは、GUI と同様の処理を実行します。

(1) バージョン収集処理

1. バージョン取得 CIM Provider のバージョン収集メソッドを実行する
2. メソッド実行結果が正常かつバージョン情報リストが存在する場合、バージョン情報リストを取得する
3. グループ名、ファイル名、ファイルバージョンを表示する
4. メソッド実行結果を表示する

バージョン情報表示のサンプルソース

```
' バージョン情報表示: prtVerInfo.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim objSnvVersionInfo
Dim beforeGroupName
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\CIMV2")

(1)
1.   ' バージョン収集メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_SnvVersion", "CollectSnvVersion")

2.   ' メソッド実行結果が正常かつバージョン情報リストが存在する場合
If objOutParams.ReturnValue = &H0 And IsNull(objOutParams.SnvVersionInfoList) = False Then
    ' バージョン情報を取得
    For Each objSnvVersionInfo In objOutParams.SnvVersionInfoList
        ' バージョン情報表示
        If objSnvVersionInfo.GroupName <> beforeGroupName Then
            Wscript.echo objSnvVersionInfo.GroupName
        End If
        Wscript.echo " " & objSnvVersionInfo.FileName & " " & objSnvVersionInfo.FileVersion
        beforeGroupName = objSnvVersionInfo.GroupName
    Next
End If
' メソッド実行結果を表示
Wscript.echo "ReturnValue CollectSnvVersion: " & Hex(objOutParams.ReturnValue)

4.
Set objSnvVersionInfo = Nothing
Set objOutParams = Nothing
Set objWMIService = Nothing
```

troubleshoot表示(CLI)

概要

GUI では、以下の処理を実行します。

1. troubleshoot ファイル有無確認

WMI のクエリを実行して、全デバイス用 CIM Provider を取得します。

デバイス用 CIM Provider のプロバイダ管理情報を取得して、接続デバイス情報がある場合、接続デバイス情報を取得します。

接続デバイス情報のモジュール情報がある場合、モジュール情報のリポジトリ管理モジュール情報を取得します。

次回適用リポジトリ管理モジュール情報が有る場合、troubleshoot ファイル名を取得します。

troubleshoot ファイル名が取得できた場合、troubleshoot ボタンを活性にし、取得できなかった場合はボタンを非活性にします。

サンプルソースの処理内容詳細

サンプルソースでは、RAID デバイスの全モジュールの troubleshoot 格納パス(絶対パス)をデバイスリストのようにツリー表示をして一括で表示します。

(1) RAID デバイスリストと RAID デバイスの troubleshoot 格納パス表示

1. RAID 用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
2. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
メソッド実行結果が異常あるいはプロバイダ管理情報の接続デバイス情報が存在しない場合、メソッド実行結果を表示して、troubleshoot 格納パス表示処理を終了する
3. WMI のクエリを実行して、PCI デバイス情報のデバイス名称(詳細)とデバイス ID を取得する
4. プロバイダ名をツリーの頂点ノードに表示する
5. 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合、PCI デバイス情報のデバイス名称(詳細)をツリー表示する
6. 接続デバイス情報の拡張情報をキーとバリューに分割して表示する
拡張情報がキーとバリューに分割できない場合は、拡張情報を表示しない
7. 接続デバイス情報のモジュール情報が存在する場合、モジュール情報を取得し、モジュール種別を表示する
8. リポジトリ管理モジュール情報を取得する

9. リポジトリ管理モジュール情報の適用種別が次回適用バージョンで、かつモジュールパスとマニュアルファイルがリポジトリ管理モジュール情報にある場合、モジュールパスとマニュアルファイル名を取得する
10. 取得したモジュールパスにマニュアルファイル名を付加して troubleshoot 格納パスを生成し、表示する
11. メソッド実行結果を表示する

troubleshoot 表示のサンプルソース

```
' troubleshoot 表示: prtTroubleshootPath.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim objPnPSet, objPnP
Dim objDevInfo, objModuleInfo, objReplInfo, objExtendedInfo
Dim strManualName, strDispManualName, strModulePath
Dim intExtendedPos
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts::" & strComputer & "\root\cimv2")

(1)
1.  ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_DevServiceRaid", "GetProviderManageInfo")
2.  ' メソッド実行結果が正常の場合
If objOutParams.ReturnValue = &H0 Then
4.  ' プロバイダ名を表示
    Wscript.Echo objOutParams.DevProviderManageInfo.ProviderName
    ' プロバイダ管理情報の接続デバイス情報が存在する場合
    If IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
        ' 接続デバイス情報を取得
        For Each objDeviceInfo In objOutParams.DevProviderManageInfo.DeviceInfo
            ' OS から PCI デバイス情報を取得
            Set objPnPSet = objWMIService.ExecQuery_
                ("SELECT Caption, DeviceID FROM Win32_PnPEntity WHERE DeviceID LIKE 'PCI%' AND
ConfigManagerErrorCode = '0' AND Status = 'OK'")
            ' 接続デバイス情報と合致する PCI デバイス情報を検索
            For Each objPnP In objPnPSet
                ' 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合
                If objDeviceInfo.DeviceId = objPnP.DeviceID Then
                    ' デバイスリストにデバイス名称を表示
                    Wscript.Echo " " & objPnP.Caption
                    ' 拡張情報が存在する場合
                    If IsNull(objDeviceInfo.ExtendedInfo) = False Then
                        ' 拡張情報を取得
                        For Each objExtendedInfo In objDeviceInfo.ExtendedInfo
                            ' 拡張情報の切り出し位置を取得
                            intExtendedPos = InStr(objExtendedInfo, "?")
                            ' 拡張情報の切り出しが可能な場合
                            If intExtendedPos <> 0 Then
                                ' 拡張情報を表示
                                Wscript.Echo " " & Mid(objExtendedInfo, 1, intExtendedPos - 1) & ":" &
Mid(objExtendedInfo, intExtendedPos + 1)
                            End If
                        Next
                    End If
                End If
                ' 接続デバイス情報のモジュール情報が存在する場合
                If IsNull(objDeviceInfo.ModuleInfo) = False Then
                    ' モジュール情報を取得
                    For Each objModuleInfo In objDeviceInfo.ModuleInfo
                        ' 表示値を初期化
                        strDispManualName = Null
                        strModulePath = Null
                        ' モジュール種別を表示
                        Wscript.echo " " "[" & objModuleInfo.ModuleKind & "]"
                    Next
                End If
            Next
        End If
    End If
End If
```


Release Note表示(CLI)

概要

GUI では、以下の処理を実行します。

1. Release Note ファイル有無確認

WMI のクエリを実行して、全デバイス用 CIM Provider を取得します。

デバイス用 CIM Provider のプロバイダ管理情報を取得して、接続デバイス情報がある場合、接続デバイス情報を取得します。

接続デバイス情報のモジュール情報がある場合、モジュール情報のリポジトリ管理モジュール情報を取得します。

次回適用リポジトリ管理モジュール情報が有る場合、Release Note ファイル名を取得します。

Release Note ファイル名が取得できた場合、Release Note ボタンを活性にし、取得できなかった場合はボタンを非活性にします。

サンプルソースの処理内容詳細

サンプルソースでは、RAID デバイスの全モジュールの Release Note 格納パス(絶対パス)をデバイスリストのようにツリー表示をして一括で表示します。

(1) RAID デバイスリストと RAID デバイスの Release Note 格納パス表示

1. RAID 用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
2. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
メソッド実行結果が異常あるいはプロバイダ管理情報の接続デバイス情報が存在しない場合、メソッド実行結果を表示して、Release Note 格納パス表示処理を終了する
3. WMI のクエリを実行して、PCI デバイス情報のデバイス名称(詳細)とデバイス ID を取得する
4. プロバイダ名をツリーの頂点ノードに表示する
5. 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合、PCI デバイス情報のデバイス名称(詳細)をツリー表示する
6. 接続デバイス情報の拡張情報をキーとバリューに分割して表示する
拡張情報がキーとバリューに分割できない場合は、拡張情報を表示しない
7. 接続デバイス情報のモジュール情報が存在する場合、モジュール情報を取得し、モジュール種別を表示する
8. リポジトリ管理モジュール情報を取得する

9. リポジトリ管理モジュール情報の適用種別が次回適用バージョンで、かつモジュールパスとリリースノートファイルがリポジトリ管理モジュール情報にある場合、モジュールパスとリリースノートファイル名を取得する
10. 取得したモジュールパスにリリースノートファイル名を付加して Release Note 格納パスを生成し、表示する
11. メソッド実行結果を表示する

Release Note 表示のサンプルソース

```
' Release Note 表示: prtReleaseNotePath.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim objPnPSet, objPnP
Dim objDeviceInfo, objModuleInfo, objRepInfo, objExtendedInfo
Dim strReleasenotesName, strDispReleasenotesName, strModulePath
Dim intExtendedPos
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts::" & strComputer & "\root\cimv2")

(1)
1.   ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_DevServiceRaid", "GetProviderManageInfo")
2.   ' メソッド実行結果が正常の場合
If objOutParams.ReturnValue = &H0 Then
4.     ' プロバイダ名を表示
    Wscript.Echo objOutParams.DevProviderManageInfo.ProviderName
    ' プロバイダ管理情報の接続デバイス情報が存在する場合
    If IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
        ' 接続デバイス情報を取得
        For Each objDeviceInfo In objOutParams.DevProviderManageInfo.DeviceInfo
            ' OS から PCI デバイス情報を取得
            Set objPnPSet = objWMIService.ExecQuery_
                ("SELECT Caption, DeviceID FROM Win32_PnPEntity WHERE DeviceID LIKE 'PCI%' AND
ConfigManagerErrorCode = '0' AND Status = 'OK'")
            ' 接続デバイス情報と合致する PCI デバイス情報を検索
            For Each objPnP In objPnPSet
                ' 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合
                If objDeviceInfo.DeviceID = objPnP.DeviceID Then
                    ' デバイスリストにデバイス名称を表示
                    Wscript.Echo " " & objPnP.Caption
                End If
                ' 拡張情報が存在する場合
                If IsNull(objDeviceInfo.ExtendedInfo) = False Then
                    ' 拡張情報を取得
                    For Each objExtendedInfo In objDeviceInfo.ExtendedInfo
                        ' 拡張情報の切り出し位置を取得
                        intExtendedPos = InStr(objExtendedInfo, "?")
                        ' 拡張情報の切り出しが可能な場合
                        If intExtendedPos <> 0 Then
                            ' 拡張情報を表示
                            Wscript.Echo " " & Mid(objExtendedInfo, 1, intExtendedPos - 1) & ":" &
Mid(objExtendedInfo, intExtendedPos + 1)
                        End If
                    Next
                End If
                ' 接続デバイス情報のモジュール情報が存在する場合
                If IsNull(objDeviceInfo.ModuleInfo) = False Then
                    ' モジュール情報を取得
                    For Each objModuleInfo In objDeviceInfo.ModuleInfo
                        ' 表示値を初期化
                        strDispReleasenotesName = Null
                        strModulePath = Null
                        ' モジュール種別を表示
                        Wscript.echo " " "[" & objModuleInfo.ModuleKind & "]"
                    Next
                End If
            Next
        End If
    End If
End If
```

```

8.          ' リポジトリ管理モジュール情報を取得
9.          For Each objReplInfo In objModuleInfo.RepositoryInfo
10.             ' 次回適用バージョンの場合
11.             If objReplInfo.ApplyKind = "update" Then
12.                 ' モジュールパスとリリースノートファイル名を取得
13.                 If IsNull(objReplInfo.ModulePath) = False And
14.                     IsNull(objReplInfo.ReleasenotesFileName) = False Then
15.                         For Each strReleasenotesName In objReplInfo.ReleasenotesFileName
16.                             strModulePath = objReplInfo.ModulePath
17.                             strDispReleasenotesName = strReleasenotesName
18.                             Exit For
19.                         Next
20.                     End If
21.                 End If
22.             Next
23.             ' 取得したモジュールパスとリリースノートファイル名を表示
24.             If IsNull(strDispReleasenotesName) = False And IsNull(strModulePath) = False Then
25.                 Wscript.echo " Release Note: " & strModulePath & "¥" & strDispReleasenotesName
26.             Else
27.                 Wscript.echo "     No Release Note. "
28.             End If
29.             Next
30.         End If
31.     Exit For
32. End If
33. Next
34. End If
35. End If
36. ' メソッド実行結果を表示
37. Wscript.echo "ReturnValue GetProviderManageInfo: " & Hex(objOutParams.ReturnValue)

Set objExtendedInfo = Nothing
Set objReplInfo = Nothing
Set objModuleInfo = Nothing
Set objDevInfo = Nothing
Set objPnP = Nothing
Set objPnPSet = Nothing
Set objOutParams = Nothing
Set objWMIService = Nothing

```

ダウンロード指示(CLI)

概要

GUI では、以下の処理を実行します。

1. ダウンロード指示

Update Manager 接続 CIM Provider を経由して、Update Manager に手動ダウンロードを実行するよう指示します。手動ダウンロード先には、CD-ROM や USB フラッシュメモリ等の外部媒体や、自動ダウンロード時の接続先である web かキャッシュサーバのいずれかを指定します。ダウンロード先指定画面表示前に自動ダウンロード設定取得処理を行い、ダウンロード先に web かキャッシュサーバが指定された場合は、取得した自動ダウンロード設定を使用してダウンロード指示メソッドを実行します。

サンプルソースの処理内容詳細

サンプルソースでは、ダウンロード元に「外部媒体」、外部媒体パスに"C:\UpdateModule\01"を指定しています。

また、ダウンロード元に「web」か「キャッシュサーバ」を指定した場合、ダウンロード指示実行の前に自動ダウンロード設定取得を行います。

(1) ダウンロード指示

1. 設定する自動ダウンロード設定情報(ダウンロード元と外部媒体パス)をダウンロード設定パラメータセットに設定する
2. ダウンロード元に web か「キャッシュサーバ」を指定した場合は自動ダウンロード設定取得メソッドを実行し、メソッド実行結果が正常の場合、戻り値から webSiteURL、プロキシ、ポート、プロキシユーザ、プロキシパスワード、キャッシュサーバパス、キャッシュサーバユーザ、キャッシュサーバパスワード、一時接続ドライブレターを取得して、ダウンロード設定パラメータセットに設定する
メソッド実行結果が異常の場合はメソッド実行結果を表示してダウンロード指示処理を終了する
3. 自動ダウンロード設定パラメータセットをダウンロード指示メソッドの引数に設定する
4. Update Manager 接続 CIM Provider のダウンロード指示メソッドを実行し、手動ダウンロードを実行する
5. メソッド実行結果を表示する

ダウンロード指示のサンプルソース

```
' ダウンロード指示: execDownload.vbs
Option Explicit

Dim objWMIService, objShare
Dim objInParam, objOutParams
Dim objSettingInParam
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\CIMV2")

(1)
' Update Manager 接続 Provider クラスを取得
Set objShare = objWMIService.Get("HITACHI_UpdateManager")
' ダウンロード指示メソッドのパラメータセットを取得
Set objInParam = objShare.Methods_("DownloadAll")._
    InParameters.SpawnInstance_()

' ダウンロード設定クラスパラメータセットを取得
Set objSettingInParam = objWMIService.Get("Hitachi_DownloadSetting").SpawnInstance_()
' ダウンロード元 0:web, 1:キャッシュサーバ、2:外部媒体
objSettingInParam.Properties_.Item("DownloadType") = 2
' 外部媒体パス ダウンロード元に'外部媒体'を指定する場合設定
objSettingInParam.Properties_.Item("ExternalMediaPath") = "C:\UpdateModule\01"
' ダウンロード元に'web'あるいは'キャッシュサーバ'を指定する場合
2. If objSettingInParam.Properties_.Item("DownloadType") = 0 Or
    objSettingInParam.Properties_.Item("DownloadType") = 1 Then
        ' 自動ダウンロード設定取得メソッドを実行して戻り値を取得
        Set objOutParams = objWMIService.ExecMethod("HITACHI_UpdateManager", "GetAutoDownloadSetting")
        ' 自動ダウンロード設定取得メソッド実行に失敗した場合
        If objOutParams.ReturnValue <> &H0 Then
            ' メソッド実行結果を表示して処理終了
            Wscript.echo "ReturnValue GetAutoDownloadSetting: " & Hex(objOutParams.ReturnValue)
            Set objSettingInParam = Nothing
            Set objOutParams = Nothing
            Set objInParam = Nothing
            Set objShare = Nothing
            Set objWMIService = Nothing
            WScript.Quit
        End If
        ' WebSiteUrl 自動ダウンロード設定を参照
        objSettingInParam.Properties_.Item("WebSiteUrl") = objOutParams.DownloadSetting.WebSiteUrl
        ' プロキシ 自動ダウンロード設定を参照
        objSettingInParam.Properties_.Item("ProxyAddress") = objOutParams.DownloadSetting.ProxyAddress
        ' ポート 自動ダウンロード設定を参照
        objSettingInParam.Properties_.Item("ProxyPort") = objOutParams.DownloadSetting.ProxyPort
        ' プロキシユーザ 自動ダウンロード設定を参照
        objSettingInParam.Properties_.Item("ProxyUser") = objOutParams.DownloadSetting.ProxyUser
        ' プロキシパスワード 自動ダウンロード設定を参照
        objSettingInParam.Properties_.Item("ProxyPassword") = objOutParams.DownloadSetting.ProxyPassword
        ' キャッシュサーバパス 自動ダウンロード設定を参照
        objSettingInParam.Properties_.Item("CacheServerPath") = objOutParams.DownloadSetting.CacheServerPath
        ' キャッシュサーバユーザ 自動ダウンロード設定を参照
        objSettingInParam.Properties_.Item("CacheServerUser") = objOutParams.DownloadSetting.CacheServerUser
        ' キャッシュサーバパスワード 自動ダウンロード設定を参照
        objSettingInParam.Properties_.Item("CacheServerPassword") =
            objOutParams.DownloadSetting.CacheServerPassword
        ' 一時接続ドライブレター 自動ダウンロード設定を参照
        objSettingInParam.Properties_.Item("TempDriveLetter") = objOutParams.DownloadSetting.TempDriveLetter
!
```

```
3. End If
   ' パラメータセットにダウンロード設定クラスパラメータセットを設定
   objInParam.Properties_.Item("DownloadSetting") = objSettingInParam

4. ' ダウンロード指示メソッドを実行して戻り値を取得
   Set objOutParams = objWMIService.ExecMethod("HITACHI_UpdateManager", "DownloadAll", objInParam)
   ' メソッド実行結果を表示
   Wscript.echo "ReturnValue DownloadAll: " & Hex(objOutParams.ReturnValue)

5. Set objSettingInParam = Nothing
   Set objOutParams = Nothing
   Set objInParam = Nothing
   Set objShare = Nothing
   Set objWMIService = Nothing
```

アップデート指示(CLI)

概要

GUI では、以下の処理を実行します。

1. アップデート指示

Update Manager 接続 CIM Provider を経由して、Update Manager に手動アップデートを実行するよう指示します。

2. システムリブート指示

アップデート指示メソッド実行結果が正常で、かつアップデート処理継続にリブートを行う必要がある場合は、リブート誘導／確認ポップアップを表示します。

リブート要否が「リブート誘導」の場合、リブート誘導ポップアップを表示します。自動リブートは行いません。

リブート要否が「強制リブート」「強制シャットダウン」の場合、リブート確認ポップアップを表示します。確認ボタン押下で、自動リブートを行います。

リブート後のアップデート処理は、Update Manager のサービス起動後に自動で再開されます。

サンプルソースの処理内容詳細

サンプルソースでは、アップデート指示メソッド実行結果が正常で、かつアップデート処理継続にリブートを行う必要がある場合、確認を行わずに自動リブートを行います。

(1) アップデート指示

1. Update Manager 接続 CIM Provider のアップデート指示メソッドを実行し、手動アップデートを実行する
2. メソッド実行結果を表示する
3. リブート要否がリブート誘導／強制リブート／強制シャットダウンの場合は、リブート要否を True、それ以外の場合はリブート要否を False と表示する

(2) システムリブート指示

1. アップデート指示メソッド実行結果が正常で、かつリブート要否がリブート誘導／強制リブート／強制シャットダウンの場合、Update Manager 接続 CIM Provider のシステムリブート指示メソッドを実行する
2. メソッド実行結果を表示する

注意事項

- アップデート処理継続にリブートを行う必要がある場合

自動リブートを行うため、他のアプリケーションの実行について、必要があれば保存したり、アプリケーションを終了させてからアップデートを行ってください。

アップデート指示のサンプルソース

```
' アップデート指示: execUpdate.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\CIMV2")

(1)
1.   ' アップデート指示メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod("HITACHI_UpdateManager", "UpdateAll")
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue UpdateAll: " & Hex(objOutParams.ReturnValue)
    ' リブート要否を表示
    If objOutParams.Reboot = &H21 Or objOutParams.Reboot = &H22 Or objOutParams.Reboot = &H23 Then
        Wscript.echo "Reboot: True"
    Else
        Wscript.echo "Reboot: False"
    End If

(2)
1.   ' メソッド実行結果が正常かつリブート要否がリブート誘導／強制リブート／強制シャットダウンの場合
    If objOutParams.ReturnValue = &H0 And (objOutParams.Reboot = &H21 Or objOutParams.Reboot = &H22 Or
    objOutParams.Reboot = &H23) Then
        ' (リブート前に確認を行う場合) 確認ポップアップを表示
        ' システムリブート指示メソッドを実行して戻り値を取得
        Set objOutParams = Nothing
        Set objOutParams = objWMIService.ExecMethod("HITACHI_UpdateManager", "ExecuteReboot")
        ' メソッド実行結果を表示
        Wscript.echo "ReturnValue ExecuteReboot: " & Hex(objOutParams.ReturnValue)
    End If

    Set objOutParams = Nothing
    Set objWMIService = Nothing
```

ロールバック指示(CLI)

概要

GUI では、以下の処理を実行します。

1. ロールバック指示

Update Manager 接続 CIM Provider を経由して、Update Manager に手動ロールバックを実行するよう指示します。

2. システムリブート指示

ロールバック指示メソッド実行結果が正常で、かつロールバック処理継続にリブートを行う必要がある場合は、リブート誘導／確認ポップアップを表示します。

リブート要否が「リブート誘導」の場合、リブート誘導ポップアップを表示します。自動リブートは行いません。

リブート要否が「強制リブート」「強制シャットダウン」の場合、リブート確認ポップアップを表示します。確認ボタン押下で、自動リブートを行います。

リブート後のロールバック処理は、Update Manager のサービス起動後に自動で再開されます。

サンプルソースの処理内容詳細

サンプルソースでは、ロールバック指示メソッド実行結果が正常で、かつロールバック処理継続にリブートを行う必要がある場合、確認を行わずに自動リブートを行います。

(1) ロールバック指示

1. Update Manager 接続 CIM Provider のロールバック指示メソッドを実行し、手動ロールバックを実行する
2. メソッド実行結果を表示する
3. リブート要否がリブート誘導／強制リブート／強制シャットダウンの場合は、リブート要否を True、それ以外の場合はリブート要否を False と表示する

(2) システムリブート指示

1. ロールバック指示メソッドの実行結果が正常で、かつリブート要否がリブート誘導／強制リブート／強制シャットダウンの場合、Update Manager 接続 CIM Provider のシステムリブート指示メソッドを実行する
2. メソッド実行結果を表示する

注意事項

- ロールバック処理継続にリブートを行う必要がある場合

自動リブートを行うため、他のアプリケーションの実行について、必要があれば保存したり、アプリケーションを終了させてからロールバックを行ってください。

ロールバック指示のサンプルソース

```
' ロールバック指示: execRollback.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts::" & strComputer & "\root\CIMV2")

(1)
1.   ' ロールバック指示メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_UpdateManager", "RollbackAll")
' メソッド実行結果を表示
Wscript.echo "ReturnValue RollbackAll: " & Hex(objOutParams.ReturnValue)
' リブート要否を表示
If objOutParams.Reboot = &H21 Or objOutParams.Reboot = &H22 Or objOutParams.Reboot = &H23 Then
    Wscript.echo "Reboot: True"
Else
    Wscript.echo "Reboot: False"
End If

(2)
1.   ' メソッド実行結果が正常かつリブート要否がリブート誘導／強制リブート／強制シャットダウンの場合
If objOutParams.ReturnValue = &H0 And (objOutParams.Reboot = &H21 Or objOutParams.Reboot = &H22 Or
objOutParams.Reboot = &H23) Then
    ' (リブート前に確認を行う場合) 確認ポップアップを表示
    ' システムリブート指示メソッドを実行して戻り値を取得
    Set objOutParams = Nothing
    Set objOutParams = objWMIService.ExecMethod("HITACHI_UpdateManager", "ExecuteReboot")
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue ExecuteReboot: " & Hex(objOutParams.ReturnValue)
End If

Set objOutParams = Nothing
Set objWMIService = Nothing
```

全デバイス用CIM Provider取得(CLI)

概要

GUI では、以下の処理を実行します。

1. 全デバイス用 CIM Provider のクラス取得

WMI のクエリを実行して、全デバイス用 CIM Provider クラスを取得します。

サンプルソースの処理内容詳細

サンプルソースでは、取得した全デバイス用 CIM Provider のクラス名を表示します。

(1) 全デバイス用 CIM Provider のクラス取得

1. WMI のクエリを実行して、全デバイス用 CIM Provider クラスを取得する
2. 取得した全デバイス用 CIM Provider クラスのクラス名を表示する

全デバイス用 CIM Provider 取得のサンプルソース

```
' 全デバイス用 CIM Provider 取得: prtDevProviderClass.vbs
Option Explicit

Dim objWMIService
Dim objClassSet, objClass
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\CIMV2")

(1)
1.   ' 全デバイス用 CIM Provider クラスを取得
Set objClassSet = objWMIService.ExecQuery("Select * From meta_class Where __SuperClass ='HITACHI_DevService'")

2.   For Each objClass In objClassSet
        ' デバイス用 CIM Provider クラス名を表示
        Wscript.Echo objClass.Path_.Class
    Next

    Set objClass = Nothing
    Set objClassSet = Nothing
    Set objWMIService = Nothing
```

全ツール用CIM Provider取得(CLI)

概要

GUI では、以下の処理を実行します。

1. 全ツール用 CIM Provider のクラス取得

WMI のクエリを実行して、全ツール用 CIM Provider クラスを取得します。

サンプルソースの処理内容詳細

サンプルソースでは、取得した全ツール用 CIM Provider のクラス名を表示します。

(1) 全ツール用 CIM Provider のクラス取得

1. WMI のクエリを実行して、全ツール用 CIM Provider クラスを取得する
2. 取得した全ツール用 CIM Provider クラスのクラス名を表示する

全ツール用 CIM Provider 取得のサンプルソース

```
' 全ツール用 CIM Provider 取得: prtToolProviderClass.vbs
Option Explicit

Dim objWMIService
Dim objClassSet, objClass
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\CIMV2")

(1)
1.   ' 全ツール用 CIM Provider クラスを取得
Set objClassSet = objWMIService.ExecQuery("Select * From meta_class Where _SuperClass = 'HITACHI_ToolService'")
For Each objClass In objClassSet
    ' ツール用 CIM Provider クラス名を表示
    Wscript.Echo objClass.Path_.Class
Next

2.   Set objClass = Nothing
     Set objClassSet = Nothing
     Set objWMIService = Nothing
```

アップデート進捗率表示 (CLI)

概要

GUI では、以下の処理を実行します。

1. アップデート進捗率の集約

WMI のクエリを実行して、全デバイス/ツール用 CIM Provider を取得します。

デバイス/ツール用 CIM Provider のプロバイダ管理情報を取得して、接続デバイス/ツール情報の有無でデバイス/ツールの接続状態を確認し、接続がある場合は接続デバイス/ツール情報を取得します。

接続デバイス/ツール情報のモジュール情報の有無を確認し、モジュール情報がある場合はアップデートステータスとアップデート進捗率を取得します。

アップデートステータスが「アップデート履歴なし」以外のモジュールのアップデート進捗率を加算し、アップデート進捗率を加算したモジュール数で割ります。

計算式 :

アップデートステータスが「アップデート履歴なし」以外のモジュールのアップデート進捗率の合計(%) ÷ アップデートステータスが「アップデート履歴なし」以外のモジュール数

計算例 : アップデート進捗率 = $(50 + 30 + 30) \div 3 = 36.67\%$

デバイス	モジュール	アップデートステータス	アップデート進捗率(%)
BIOS/EFI	Firmware	アップデート完了／リブート待ち	50
BMC	Firmware	アップデート履歴なし	0
RAID	Firmware	アップデート中	30
	Driver	アップデート履歴なし	0
	Utility	アップデート前処理完了	30

参考 : アップデートステータスとアップデート進捗率

ステータス (16 進数)	ステータス文言	ステータス	アップデート 進捗率(%)
0	There is no update history.	アップデート履歴なし	0
10	Executing backup	バックアップ中	0
11	Backup completion	バックアップ完了	25
12	Backup failure	バックアップ失敗	100
20	Executing update preprocessing	アップデート前処理中	25
21	Waiting for reboot.(Update preprocessing)	アップデート前処理完了／リブート待ち	30
22	Update preprocessing completion.	アップデート前処理完了	30

ステータス (16進数)	ステータス文言	ステータス	アップデート 進捗率(%)
23	Update preprocessing failure	アップデート前処理失敗	100
30	Executing update	アップデート中	30
31	Waiting for reboot.(Update)	アップデート完了／リブート待ち	50
32	Update completion	アップデート完了	50
33	Update failure	アップデート失敗	100
40	Executing restore	リストア中	50
41	Waiting for reboot.(Restore)	リストア完了／リブート待ち	75
42	Restore completion	リストア完了	75
50	Executing an application	アップデート適用判定中	75
51	Application success	アップデート適用成功	100
52	Application failure	アップデート適用失敗	100

サンプルソースの処理内容詳細

サンプルソースでは、RAID デバイスの全モジュールのアップデートステータスとアップデート進捗率をデバイスリストのようにツリー表示をして一括で表示します。

- (1) RAID デバイスリストと RAID デバイスのアップデートステータス・アップデート進捗率表示
 1. RAID 用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
 2. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
 - メソッド実行結果が異常あるいはプロバイダ管理情報の接続デバイス情報が存在しない場合、メソッド実行結果を表示して RAID デバイスリストと RAID デバイスのアップデートステータス・アップデート進捗率表示処理を終了する
 3. WMI のクエリを実行して、PCI デバイス情報のデバイス名称(詳細)とデバイス ID を取得する
 4. プロバイダ名を表示する
 5. 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合、PCI デバイス情報のデバイス名称(詳細)をツリー表示する
 6. 接続デバイス情報の拡張情報をキーとバリューに分割して表示する
拡張情報がキーとバリューに分割できない場合は、拡張情報を表示しない
 7. 接続デバイス情報のモジュール情報が存在する場合、モジュール情報を取得し、モジュール種別、アップデートステータス、アップデート進捗率を表示する
 8. メソッド実行結果を表示する

アップデート進捗率表示のサンプルソース

```
' アップデート進捗率表示(進捗確認): prtUpdateProgress.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim objPnPSet, objPnP
Dim objDeviceInfo, objModuleInfo, objExtendedInfo
Dim intExtendedPos
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts::" & strComputer & "\root\cimv2")

(1)
1.   ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_DevServiceRaid", "GetProviderManageInfo")
2.   ' メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合
If objOutParams.ReturnValue = &H0 And IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
    ' OS から PCI デバイス情報を取得
    Set objPnPSet = objWMIService.ExecQuery_
    ("SELECT Caption, DeviceID FROM Win32_PnPEntity WHERE DeviceID LIKE 'PCI%' AND ConfigManagerErrorCode = '0'
    AND Status = 'OK'")
4.   ' デバイス名を表示
    Wscript.Echo objOutParams.DevProviderManageInfo.ProviderName
    ' 接続デバイス情報と合致する PCI デバイス情報を検索
    For Each objDeviceInfo In objOutParams.DevProviderManageInfo.DeviceInfo
        For Each objPnP In objPnPSet
            ' 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合
            If objDeviceInfo.DeviceId = objPnP.DeviceID Then
                ' デバイスリストにデバイス名称を表示
                Wscript.Echo " " & objPnP.Caption
                ' 拡張情報が存在する場合
                If IsNull(objDeviceInfo.ExtendedInfo) = False Then
                    ' 拡張情報を取得
                    For Each objExtendedInfo In objDeviceInfo.ExtendedInfo
                        ' 拡張情報の切り出し位置を取得
                        intExtendedPos = InStr(objExtendedInfo, "?")
                        ' 拡張情報の切り出しが可能な場合
                        If intExtendedPos <> 0 Then
                            ' 拡張情報を表示
                            Wscript.Echo " " & Mid(objExtendedInfo, 1, intExtendedPos - 1) & ":" &
                            Mid(objExtendedInfo, intExtendedPos + 1)
                        End If
                    Next
                End If
            End If
            ' 接続デバイス情報のモジュール情報が存在する場合
            If IsNull(objDeviceInfo.ModuleInfo) = False Then
                ' モジュール情報を取得
                For Each objModuleInfo In objDeviceInfo.ModuleInfo
                    ' モジュール種別を表示
                    Wscript.echo "[" & objModuleInfo.ModuleKind & "]"
                    ' アップデートステータスを表示
                    Wscript.echo " UpdateStatus: " & Hex(objModuleInfo.UpdateStatus)
                    ' アップデート進捗率を表示
                    Wscript.echo " UpdateProgress: " & objModuleInfo.UpdateProgress
                Next
            End If
        Exit For
    End If

```

```
        End If
    Next
    Next
End If
' メソッド実行結果を表示
Wscript.echo "ReturnValue GetProviderManageInfo: " & Hex(objOutParams.ReturnValue)

Set objExtendedInfo = Nothing
Set objModuleInfo = Nothing
Set objDeviceInfo = Nothing
Set objPnP = Nothing
Set objPnPSet = Nothing
Set objOutParams = Nothing
Set objWMIService = Nothing
```

自動ダウンロード設定取得(CLI)

概要

GUI では、以下の処理を実行します。

1. 自動ダウンロード設定取得

Update Manager で管理している、自動ダウンロードの可否や実行日時等の自動ダウンロードスケジュール情報および HTTP 接続時のプロキシ設定等の自動ダウンロード設定情報を Update Manager 接続 CIM Provider を経由して取得します。

サンプルソースの処理内容詳細

サンプルソースでは、GUI と同様の処理を実行します。

(1) 自動ダウンロード設定取得

1. Update Manager 接続 CIM Provider の自動ダウンロード設定取得メソッドを実行し、自動ダウンロード設定を取得する
2. メソッド実行結果が正常の場合、戻り値の自動ダウンロードスケジュール情報と自動ダウンロード設定情報を取得して表示する
3. メソッド実行結果を表示する

自動ダウンロード設定取得のサンプルソース

```
' 自動ダウンロード設定取得: prtDownloadSetting.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\CIMV2")

(1)
1.   ' 自動ダウンロード設定取得メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_UpdateManager", "GetAutoDownloadSetting")
' メソッド実行結果が正常の場合
If objOutParams.ReturnValue = &H0 Then
    ' 自動ダウンロードスケジュールを表示
    ' 自動更新
    Wscript.echo "DownloadSchedule IsAutoSetting: " & objOutParams.DownloadSchedule.IsAutoSetting
    ' 繰り返し
    Wscript.echo "DownloadSchedule RepeatType: " & objOutParams.DownloadSchedule.RepeatType
    ' 日付
    Wscript.echo "DownloadSchedule DayOfWeek: " & objOutParams.DownloadSchedule.DayOfWeek
    ' 曜日
    Wscript.echo "DownloadSchedule Date: " & objOutParams.DownloadSchedule.Date
    ' 時刻
    Wscript.echo "DownloadSchedule Time: " & objOutParams.DownloadSchedule.Time
    ' 自動ダウンロード設定を表示
    ' ダウンロード元
    Wscript.echo "DownloadSetting DownloadType: " & objOutParams.DownloadSetting.DownloadType
    ' WebSiteUrl
    Wscript.echo "DownloadSetting WebSiteUrl: " & objOutParams.DownloadSetting.WebSiteUrl
    ' プロキシ
    Wscript.echo "DownloadSetting ProxyAddress: " & objOutParams.DownloadSetting.ProxyAddress
    ' ポート
    Wscript.echo "DownloadSetting ProxyPort: " & objOutParams.DownloadSetting.ProxyPort
    ' プロキシユーザ
    Wscript.echo "DownloadSetting ProxyUser: " & objOutParams.DownloadSetting.ProxyUser
    ' プロキシパスワード
    Wscript.echo "DownloadSetting ProxyPassword: " & objOutParams.DownloadSetting.ProxyPassword
    ' キャッシュサーバパス
    Wscript.echo "DownloadSetting CacheServerPath: " & objOutParams.DownloadSetting.CacheServerPath
    ' キャッシュサーバユーザ
    Wscript.echo "DownloadSetting CacheServerUser: " & objOutParams.DownloadSetting.CacheServerUser
    ' キャッシュサーバパスワード
    Wscript.echo "DownloadSetting CacheServerPassword: " & objOutParams.DownloadSetting.CacheServerPassword
    ' 一時接続ドライブレター
    Wscript.echo "DownloadSetting TempDriveLetter: " & objOutParams.DownloadSetting.TempDriveLetter
End If
' メソッド実行結果を表示
Wscript.echo "ReturnValue GetAutoDownloadSetting: " & Hex(objOutParams.ReturnValue)

Set objOutParams = Nothing
Set objWMIService = Nothing
```

自動ダウンロード設定更新(CLI)

概要

GUI では、以下の処理を実行します。

1. 自動ダウンロード設定更新

Update Manager で管理している、自動ダウンロードの可否や実行日時等の自動ダウンロードスケジュール情報および HTTP 接続時のプロキシ設定等の自動ダウンロード設定情報を Update Manager 接続 CIM Provider を経由して更新します。

サンプルソースの処理内容詳細

サンプルソースでは、GUI と同様の処理を実行します。

(1) 自動ダウンロード設定更新

1. 設定する自動ダウンロードスケジュール情報をダウンロードスケジュールパラメータセットに設定する
2. 設定する自動ダウンロード設定情報をダウンロード設定パラメータセットに設定する
3. ダウンロードスケジュールパラメータセットとダウンロード設定パラメータセットを自動ダウンロード設定更新メソッドの引数に設定する
4. Update Manager 接続 CIM Provider の自動ダウンロード設定更新メソッドを実行し、自動ダウンロード設定を更新する
5. メソッド実行結果を表示する

自動ダウンロード設定更新のサンプルソース

```
' 自動ダウンロード設定更新: setDownloadSetting.vbs
Option Explicit

Dim objWMIService, objShare
Dim objInParam, objOutParams
Dim objScheduleInParam, objSettingInParam
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\CIMV2")

(1)
' Update Manager 接続 Provider クラスを取得
Set objShare = objWMIService.Get("HITACHI_UpdateManager")
' 自動ダウンロード設定更新メソッドのパラメータセットを取得
Set objInParam = objShare.Methods_("SetAutoDownloadSetting").inParameters.SpawnInstance_()
' ダウンロードスケジュールクラスパラメータセットを取得
Set objScheduleInParam = objWMIService.Get("Hitachi_DownloadSchedule").SpawnInstance_()
' 自動更新 True:する, False:しない
objScheduleInParam.Properties_.Item("IsAutoSetting") = False
' 繰り返し 0:日付指定・繰り返しなし, 1:曜日指定・繰り返しなし
objScheduleInParam.Properties_.Item("RepeatType") = 0
' 日付 "01"~"31"
objScheduleInParam.Properties_.Item("Date") = "01"
' 曜日 "0":日曜日, "1":月曜日, "2":火曜日, "3":水曜日, "4":木曜日, "5":金曜日, "6":土曜日
objScheduleInParam.Properties_.Item("DayOfWeek") = "0"
' 時刻 "0000"~"2359"
objScheduleInParam.Properties_.Item("Time") = "0000"
' パラメータセットにダウンロードスケジュールクラスパラメータセットを設定
objInParam.Properties_.Item("DownloadSchedule") = objScheduleInParam
' ダウンロード設定クラスパラメータセットを取得
Set objSettingInParam = objWMIService.Get("Hitachi_DownloadSetting").SpawnInstance_()
' ダウンロード元 0:web, 1:キャッシュサーバ
objSettingInParam.Properties_.Item("DownloadType") = 0
' WebSiteUrl
objSettingInParam.Properties_.Item("WebSiteUrl") = "http://www2.hitachi.co.jp/ubs/pc/ha/download/snv/"
' プロキシ
objSettingInParam.Properties_.Item("ProxyAddress") = ""
' ポート
objSettingInParam.Properties_.Item("ProxyPort") = ""
' プロキシユーザー
objSettingInParam.Properties_.Item("ProxyUser") = ""
' プロキシパスワード
objSettingInParam.Properties_.Item("ProxyPassword") = ""
' キャッシュサーバパス
objSettingInParam.Properties_.Item("CacheServerPath") = ""
' キャッシュサーバユーザー
objSettingInParam.Properties_.Item("CacheServerUser") = ""
' キャッシュサーバパスワード
objSettingInParam.Properties_.Item("CacheServerPassword") = ""
' 一時接続ドライブレター
objSettingInParam.Properties_.Item("TempDriveLetter") = ""
' パラメータセットにダウンロード設定クラスパラメータセットを設定
objInParam.Properties_.Item("DownloadSetting") = objSettingInParam
' 自動ダウンロード設定更新メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_UpdateManager", "SetAutoDownloadSetting", objInParam)
' メソッド実行結果を表示
Wscript.echo "ReturnValue SetAutoDownloadSetting: " & Hex(objOutParams.ReturnValue)
```

```
Set objSettingInParam = Nothing
Set objScheduleInParam = Nothing
Set objOutParams = Nothing
Set objInParam = Nothing
Set objShare = Nothing
Set objWMIService = Nothing
```

自動アップデート設定取得(CLI)

概要

GUI では、以下の処理を実行します。

1. 自動アップデート設定取得

Update Manager で管理している、自動アップデートの可否や実行日時等の自動アップデートスケジュール情報を Update Manager 接続 CIM Provider を経由して取得します。

サンプルソースの処理内容詳細

サンプルソースでは、GUI と同様の処理を実行します。

(1) 自動アップデート設定取得

1. Update Manager 接続 CIM Provider の自動アップデート設定取得メソッドを実行し、自動アップデート設定を取得する
2. メソッド実行結果が正常の場合、戻り値の自動アップデートスケジュール情報を表示する
3. メソッド実行結果を表示する

自動アップデート設定取得のサンプルソース

```
' 自動アップデート設定取得: prtUpdateSetting.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\CIMV2")

(1)
1.   ' 自動アップデート設定取得メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_UpdateManager", "GetAutoUpdateSetting")
' メソッド実行結果が正常の場合
2.   If objOutParams.ReturnValue = &H0 Then
    ' 自動更新
    Wscript.echo "UpdateSchedule IsAutoSetting: " & objOutParams.UpdateSchedule.IsAutoSetting
    ' 繰り返し
    Wscript.echo "UpdateSchedule RepeatType: " & objOutParams.UpdateSchedule.RepeatType
    ' 日付
    Wscript.echo "UpdateSchedule DayOfWeek: " & objOutParams.UpdateSchedule.DayOfWeek
    ' 曜日
    Wscript.echo "UpdateSchedule Date: " & objOutParams.UpdateSchedule.Date
    ' 時刻
    Wscript.echo "UpdateSchedule Time: " & objOutParams.UpdateSchedule.Time
End If
' メソッド実行結果を表示
Wscript.echo "ReturnValue GetAutoUpdateSetting: " & Hex(objOutParams.ReturnValue)

3.

Set objOutParams = Nothing
Set objWMIService = Nothing
```

自動アップデート設定更新(CLI)

概要

GUI では、以下の処理を実行します。

1. 自動アップデート設定更新

Update Manager で管理している、自動アップデートの可否や実行日時等の自動アップデートスケジュール情報を Update Manager 接続 CIM Provider を経由して更新します。

サンプルソースの処理内容詳細

サンプルソースでは、GUI と同様の処理を実行します。

(1) 自動アップデート設定更新

1. 設定する自動アップデートスケジュール情報をアップデートスケジュールパラメータセットに設定する
2. アップデートスケジュールパラメータセットを自動アップデート設定更新メソッドの引数に設定する
3. Update Manager 接続 CIM Provider の自動アップデート設定更新メソッドを実行し、自動アップデート設定を更新する
4. メソッド実行結果を表示する

自動アップデート設定更新のサンプルソース

```
' 自動アップデート設定更新: setUpdateSetting.vbs
Option Explicit

Dim objWMIService, objShare
Dim objInParam, objOutParams
Dim objScheduleInParam
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root\cimv2 に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\CIMV2")

(1)
' Update Manager 接続 Provider クラスを取得
Set objShare = objWMIService.Get("HITACHI_UpdateManager")
' 自動アップデート設定更新メソッドのパラメータセットを取得
Set objInParam = objShare.Methods_("SetAutoUpdateSetting").inParameters.SpawnInstance_()
' アップデートスケジュールクラスパラメータセットを取得
Set objScheduleInParam = objWMIService.Get("Hitachi_UpdateSchedule").SpawnInstance_()
' 自動更新 True:する, False:しない
objScheduleInParam.Properties_.Item("IsAutoSetting") = False
' 繰り返し 0:日付指定・繰り返しなし, 1:曜日指定・繰り返しなし
objScheduleInParam.Properties_.Item("RepeatType") = 0
' 日付 "01"~"31"
objScheduleInParam.Properties_.Item("Date") = "01"
' 曜日 "0":日曜日, "1":月曜日, "2":火曜日, "3":水曜日, "4":木曜日, "5":金曜日, "6":土曜日
objScheduleInParam.Properties_.Item("DayOfWeek") = "0"
' 時刻 "0000"~"2359"
objScheduleInParam.Properties_.Item("Time") = "0000"
' パラメータセットにアップデートスケジュールクラスパラメータセットを設定
objInParam.Properties_.Item("UpdateSchedule") = objScheduleInParam
' 自動アップデート設定更新メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_UpdateManager", "SetAutoUpdateSetting", objInParam)
' メソッド実行結果を表示
Wscript.echo "ReturnValue SetAutoUpdateSetting: " & Hex(objOutParams.ReturnValue)

Set objScheduleInParam = Nothing
Set objOutParams = Nothing
Set objInParam = Nothing
Set objShare = Nothing
Set objWMIService = Nothing
```

トレースログ取得(CLI)

概要

GUI では、以下の処理を実行します。

1. トレースログ取得

ログ収集 CIM Provider から、Update Manager や CIM Provider が出力したトレースログを取得します。ログは、ログ出力日時の降順にソートされます。

ログ収集開始日時を指定した場合、指定日時以降のトレースログを最大 200 行取得します。指定が無い場合は、直近のトレースログを 200 行取得します。

サンプルソースの処理内容詳細

サンプルソースでは、ログ収集開始日時を指定してトレースログを取得します。

(1) トレースログ取得

1. ログ収集開始日時を処理結果収集メソッドの引数に設定する
2. ログ収集 Provider の処理結果ログ収集メソッドを実行し、ログリストを取得する
3. メソッド実行結果が正常かつログリストが存在する場合、ログリストを表示する
4. メソッド実行結果を表示する

トレースログ取得のサンプルソース

```
' トレースログ取得: prtTraceLog.vbs
Option Explicit

Dim objWMIService, objShare
Dim objInParam, objOutParams
Dim LogMessage
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\CIMV2")

(1)
' ログ収集 Provider クラスを取得
Set objShare = objWMIService.Get("HITACHI_LogManager")
' 処理結果ログ収集メソッドのパラメータセットを取得
Set objInParam = objShare.Methods_("CollectResultLog").inParameters.SpawnInstance_()
' パラメータセットにパラメータを設定
' ログ収集開始日時 (yyyy/mm/dd hh:mm:ss 形式) 指定日時以降のログを収集したい場合設定
objInParam.Properties_.Item("FromDateTime") = "2012/06/01 10:00:00"
1.   ' 処理結果ログ収集メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_LogManager", "CollectResultLog", objInParam)
' メソッド実行結果が正常かつ処理結果ログが存在する場合
3.   If objOutParams.ReturnValue = &H0 And IsNull(objOutParams.LogList) = False Then
    ' 処理結果ログを表示
    Wscript.echo "ResultLog: "
    For Each LogMessage In objOutParams.LogList
        Wscript.echo LogMessage
    Next
End If
4.   ' メソッド実行結果を表示
Wscript.echo "ReturnValue CollectResultLog: " & Hex(objOutParams.ReturnValue)

Set objOutParams = Nothing
Set objInParam = Nothing
Set objShare = Nothing
Set objWMIService = Nothing
```

更新適用重要度表示(CLI)

概要

GUI では、以下の処理を実行します。

1. デバイス・ツール用 CIM Provider 更新適用重要度表示

ダウンロードを許可する重要度を各デバイス・ツール用 CIM Provider で管理しているプロバイダ管理情報から取得します。

ダウンロードを許可する重要度は、各 CIM Provider で管理しているデバイス毎に異なります。各デバイスのモジュール(firm、driver、util)のうち最初に取得したモジュール情報の重要度を取得します。

サンプルソースの処理内容詳細

サンプルソースでは、デバイス用 CIM Provider 及びツール用 CIM Provider の更新適用重要度を取得します。

また、取得した更新適用重要度をデバイスリストのようにツリー表示を行います。

(1) システムデバイス更新適用重要度表示

1. BIOS/EFI 用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する

2. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する

メソッド実行結果が異常の場合、メソッド実行結果を表示して BMC 用 CIM Provider のプロバイダ管理情報を取得する

3. システムデバイス名(固定文言:System)をツリーの頂点ノードに表示する

4. 接続デバイス情報からモジュール情報を取得し、モジュール情報から更新適用重要度を取得してツリー表示する

5. メソッド実行結果を表示する

6. BMC 用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する

7. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する

メソッド実行結果が異常の場合、メソッド実行結果を表示して次のデバイス用 CIM Provider のプロバイダ管理情報を取得する

8. 接続デバイス情報からモジュール情報を取得し、モジュール情報から更新適用重要度を取得してツリー表示する

9. メソッド実行結果を表示する

(2) システムデバイス更新適用重要度表示

1. WMI のクエリを実行して全デバイス用 CIM Provider クラスを取得する
2. BIOS/EFI と BMC 以外の全デバイス用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
3. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
メソッド実行結果が異常の場合、メソッド実行結果を表示して次のデバイス用 CIM Provider のプロバイダ管理情報を取得する
4. WMI のクエリを実行して、PCI デバイス情報のデバイス名称(詳細)とデバイス ID を取得する
5. プロバイダ名をツリーの頂点ノードに表示する
6. 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合、PCI デバイス情報のデバイス名称(詳細)をツリー表示する
7. 接続デバイス情報の拡張情報をキーとバリューに分割して表示する
拡張情報がキーとバリューに分割できない場合は、拡張情報を表示しない
8. 接続デバイス情報からモジュール情報を取得し、モジュール情報から更新適用重要度を取得してツリー表示する
9. メソッド実行結果を表示する

(3) ツール更新適用重要度表示

1. WMI のクエリを実行して全ツール用 CIM Provider クラスを取得する
2. 全ツール用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
3. メソッド実行結果が正常かつプロバイダ管理情報の接続ツール情報が存在する場合、接続ツール情報を取得する
メソッド実行結果が異常の場合、メソッド実行結果を表示して次のツール用 CIM Provider のプロバイダ管理情報を取得する
4. ツールリスト名(固定文言: Tool)をツリーの頂点ノードに表示する
5. プロバイダ名をツリー表示する
6. 接続デバイス情報からモジュール情報を取得し、モジュール情報から更新適用重要度を取得してツリー表示する
7. メソッド実行結果を表示する

更新適用重要度表示のサンプルソース

```
' 更新適用重要度表示: prtUpdatePriority.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim objClassSet, objClass
Dim objPnPSet, objPnP
Dim objDevToolInfo, objModuleInfo, objExtendedInfo
Dim intExtendedPos
Dim ProviderList(1), strProvider
Dim strComputer
Const BiosProvider = "HITACHI_DevServiceBios"
Const BMCProvider = "HITACHI_DevServiceBMC"

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

(1)      ' システムデバイス更新適用重要度表示
' システムデバイスプロバイダリストを作成
ProviderList(0) = BiosProvider
ProviderList(1) = BMCProvider

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts::" & strComputer & "\root\CIMV2")

' システムデバイスツリー表示
Wscript.echo "System"
For Each strProvider In ProviderList
    ' BIOS/EFI 用 CIM Provider の場合
    If strProvider = BiosProvider Then
        ' BIOS/EFI デバイスツリー表示
        Wscript.echo " BIOS/EFI"
    ' BMC 用 CIM Provider の場合
    ElseIf strProvider = BMCProvider Then
        ' BMC デバイスツリー表示
        Wscript.echo " BMC"
    End If
    ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod(strProvider, "GetProviderManageInfo")
    ' メソッド実行結果が正常の場合
    If objOutParams.ReturnValue = &H0 Then
        ' プロバイダ管理情報の接続デバイス情報が存在する場合
        If IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
            ' 接続デバイス情報を取得
            For Each objDevToolInfo In objOutParams.DevProviderManageInfo.DeviceInfo
                ' システムデバイス更新適用重要度をツリー表示
                DisplayUpdatePriority()
            Next
            ' メソッド実行結果を表示
            Wscript.echo " ReturnValue GetProviderManageInfo (" & Replace(strProvider, "HITACHI_DevService", "") & ") : " & Hex(objOutParams.ReturnValue)
        End If
        ' メソッド実行結果が異常の場合
        Else
            ' メソッド実行結果を表示
            Wscript.echo " ReturnValue GetProviderManageInfo (" & Replace(strProvider, "HITACHI_DevService", "") & ") : " & Hex(objOutParams.ReturnValue)
        End If
    End If
Next
!
```

```

(2)
1.      ' デバイス更新適用重要度表示
2.      ' 全デバイス用 CIM Provider クラスを取得
3.      Set objClassSet = objWMIService.ExecQuery("Select * From meta_class Where __SuperClass = 'HITACHI_DevService'")

4.      For Each objClass In objClassSet
5.          ' BIOS/EFI 用 CIM Provider でも BMC 用 CIM Provider でもない場合
6.          If objClass.Path_.Class <> BiosProvider And objClass.Path_.Class <> BMCProvider Then
7.              ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
8.              Set objOutParams = objWMIService.ExecMethod(objClass.Path_.Class, "GetProviderManageInfo")
9.              ' メソッド実行結果が正常の場合
10.             If objOutParams.ReturnValue = &H0 Then
11.                 ' プロバイダ管理情報の接続デバイス情報が存在する場合
12.                 If IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
13.                     ' OS から PCI デバイス情報を取得
14.                     Set objPnPSet = objWMIService.ExecQuery(
15.                         ("SELECT Caption,DeviceID FROM Win32_PnPEntity WHERE DeviceID LIKE 'PCI%' AND
16. ConfigManagerErrorCode = '0' AND Status = 'OK'"))
17.                     ' プロバイダ名を表示
18.                     Wscript.Echo objOutParams.DevProviderManageInfo.ProviderName
19.                     ' 接続デバイス情報を取得
20.                     For Each objDevToolInfo In objOutParams.DevProviderManageInfo.DeviceInfo
21.                         ' 接続デバイス情報と合致する PCI デバイス情報を検索
22.                         For Each objPnP In objPnPSet
23.                             ' 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合
24.                             If objDevToolInfo.DeviceID = objPnP.DeviceID Then
25.                                 ' デバイス情報をツリー表示
26.                                 ' デバイス名称を表示
27.                                 Wscript.Echo " " & objPnP.Caption
28.                                 ' 拡張情報が存在する場合
29.                                 If IsNull(objDevToolInfo.ExtendedInfo) = False Then
30.                                     ' 拡張情報を取得
31.                                     For Each objExtendedInfo In objDevToolInfo.ExtendedInfo
32.                                         ' 拡張情報の切り出し位置を取得
33.                                         intExtendedPos = InStr(objExtendedInfo, "?")
34.                                         ' 拡張情報の切り出しが可能な場合
35.                                         If intExtendedPos <> 0 Then
36.                                             ' 拡張情報を表示
37.                                             Wscript.Echo " " & Mid(objExtendedInfo, 1, intExtendedPos - 1) & ":" & Mid(objExtendedInfo, intExtendedPos + 1)
38.                                         End If
39.                                         Next
40.                                     End If
41.                                     ' デバイス更新適用重要度をツリー表示
42.                                     DisplayUpdatePriority()
43.                                     Exit For
44.                                 End If
45.                                 Next
46.                             End If
47.                         Next
48.                     Next
49.                     ' メソッド実行結果を表示
50.                     Wscript.echo "ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
51. "HITACHI_DevService", "") & "):" & Hex(objOutParams.ReturnValue)
52.                 End If
53.                 ' メソッド実行結果が異常の場合
54.                 Else
55.                     ' メソッド実行結果を表示
56.                     Wscript.echo "ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
57. "HITACHI_DevService", "") & "):" & Hex(objOutParams.ReturnValue)
58.                 End If
59.             End If
60.         Next

(3)
1.      ' ツールアップデート情報表示
2.      ' 全ツール用 CIM Provider クラスを取得
3.      Set objClassSet = objWMIService.ExecQuery("Select * From meta_class Where __SuperClass = 'HITACHI_ToolService'")

```

```

4.      ' ツールツリー表示
Wscript.echo "Tool"
For Each objClass In objClassSet
    ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod(objClass.Path_.Class, "GetProviderManageInfo")
    ' メソッド実行結果が正常の場合
    If objOutParams.ReturnValue = &H0 Then
        ' プロバイダ管理情報の接続ツール情報が存在する場合
        If IsNull(objOutParams.ToolProviderManageInfo.ToolInfo) = False Then
            ' プロバイダ名を表示
            Wscript.Echo "" & objOutParams.ToolProviderManageInfo.ProviderName
            ' 接続ツール情報を取得
            For Each objDevToolInfo In objOutParams.ToolProviderManageInfo.ToolInfo
                ' ツール更新適用重要度をツリー表示
                DisplayUpdatePriority()
            Next
            ' メソッド実行結果を表示
            Wscript.echo " ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
"HTACHI_ToolService", "") & "): " & Hex(objOutParams.ReturnValue)
        End If
        ' メソッド実行結果が異常の場合
    Else
        ' メソッド実行結果を表示
        Wscript.echo " ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
"HTACHI_ToolService", "") & "): " & Hex(objOutParams.ReturnValue)
    End If
    Next

    Set objExtendedInfo = Nothing
    Set objModuleInfo = Nothing
    Set objDevToolInfo = Nothing
    Set objPnP = Nothing
    Set objPnPSet = Nothing
    Set objClass = Nothing
    Set objClassSet = Nothing
    Set objOutParams = Nothing
    Set objWMIService = Nothing

    ' /////////////////////////////////
    ' 更新適用重要度表示
    ' ///////////////////////////////
Sub DisplayUpdatePriority()

    ' 接続デバイス・ツール情報のモジュール情報が存在する場合
    If IsNull(objDevToolInfo.ModuleInfo) = False Then
        ' モジュール情報を取得
        For Each objModuleInfo In objDevToolInfo.ModuleInfo
            ' モジュール種別を表示
            Wscript.echo "[" & objModuleInfo.ModuleKind & "]"
            ' 更新適用重要度を表示
            Wscript.echo " UpdatePriority: " & objModuleInfo.UpdatePriority
        Next
    End If
End Sub

```

更新適用重要度設定(CLI)

概要

GUI では、以下の処理を実行します。

1. デバイス・ツール用 CIM Provider 更新適用重要度設定

ダウンロードを許可する重要度を各デバイス・ツール用 CIM Provider に設定します。ダウンロードを許可する重要度は、各 CIM Provider で管理しているデバイス単位に設定します。

重要度設定処理を行うため、処理の前に各デバイス・ツール用 CIM Provider の操作権限を取得し、処理の後に権限を解放します。

サンプルソースの処理内容詳細

サンプルソース 1 では、デバイス用 CIM Provider 及びツール用 CIM Provider の更新適用重要度を設定します。(同一の更新適用重要度を一括設定します)

(1) 全デバイス・ツール用 CIM Provider 操作権限取得

1. WMI のクエリを実行して全デバイス・ツール用 CIM Provider クラスを取得する
 2. アプリケーションキーに"CLI"を設定して、全デバイス・ツール用 CIM Provider の操作権限取得メソッドを実行し、操作権限を取得する
- メソッド実行結果が異常の場合、メソッド実行結果を表示して、操作権限取得済のデバイス・ツール用 CIM Provider に対して取得順と逆順で操作権限解放メソッドを実行して、操作権限を解放、更新適用重要度設定処理を終了する

(2) デバイス用 CIM Provider 更新適用重要度設定

1. デバイス用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
 2. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
- メソッド実行結果が異常の場合、メソッド実行結果を表示して次のデバイス用 CIM Provider のプロバイダ管理情報を取得する
3. 接続デバイス情報からデバイス ID を取得する
 4. アプリケーションキーに"CLI"を重要度に 1 をデバイス ID に接続デバイス情報から取得したデバイス ID を設定して全モジュール種別に対して更新適用重要度設定メソッドを実行し、更新適用重要度を設定する
 5. メソッド実行結果を表示する

(3) ツール用 CIM Provider 更新適用重要度設定

1. アプリケーションキーに"CLI"を重要度に 1 を設定して更新適用重要度設定メソッドを実行し、更新適用重要度を設定する
2. メソッド実行結果を表示する

(4) 全デバイス・ツール用 CIM Provider 操作権限解放

1. アプリケーションキーに"CLI"を設定して、全デバイス・ツール用 CIM Provider に対して操作権限取得順と逆順で操作権限解放メソッドを実行して、操作権限を解放する
2. メソッド実行結果が異常の場合、メソッド実行結果を表示する

サンプルソース 2 では、RAID 用 CIM Provider の更新適用重要度を設定します。(RAID 用 CIM Provider の更新適用重要度を個別設定します)

(1) RAID 用 CIM Provider 操作権限取得

1. アプリケーションキーに"CLI"を設定して、RAID 用 CIM Provider の操作権限取得メソッドを実行し、操作権限を取得する
- メソッド実行結果が異常の場合、メソッド実行結果を表示し、操作権限解放メソッドを実行して、操作権限を解放、更新適用重要度設定処理を終了する

(2) RAID 用 CIM Provider 更新適用重要度設定

1. RAID 用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
2. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
- メソッド実行結果が異常の場合、メソッド実行結果を表示して更新適用重要度設定処理を終了する
3. 接続デバイス情報からデバイス ID を取得する
4. アプリケーションキーに"CLI"を重要度に 1 をデバイス ID に接続デバイス情報から取得したデバイス ID を設定して全モジュール種別に対して更新適用重要度設定メソッドを実行し、更新適用重要度を設定する
5. メソッド実行結果を表示する

(3) RAID 用 CIM Provider 操作権限解放

1. アプリケーションキーに"CLI"を設定して、RAID 用 CIM Provider に対して操作権限解放メソッドを実行して、操作権限を解放する
2. メソッド実行結果が異常の場合、メソッド実行結果を表示する

● UpdateManage 操作権限解放処理結果が異常の場合



操作権限解放処理に失敗した場合、失敗したデバイスは操作権限を取得したままとなります。

Server Navigator の他の処理が実行不可能となりますので、メソッド実行結果と同時に表示したデバイス名を確認し、Server Navigator の他の処理が実行中でないことを確認した上で、該当デバイスの「Hitachi Server Navigator ユーザーズガイド アップデート機能」 - 「トラブルシューティング」の「ステータス回復スクリプト：操作権限クリア」を実行して、操作権限の解放を行ってください。

-
- 重要度設定処理時に設定するアプリケーションキーについて
設定するアプリケーションキーは、Server Navigator の他の処理との競合を防ぐため、"CLI"を設定してください。
 - 更新適用重要度設定メソッド(デバイス用 CIM Provider)のパラメータについて
誤ったデバイス ID やモジュール種別を設定して更新適用重要度設定を行うと設定処理が行われないため、設定値に誤りがないことを確認してください。
更新適用重要度設定はデバイス単位での管理のため、必ず全モジュール種別に対して更新適用重要度設定を行ってください。
-

更新適用重要度設定のサンプルソース 1

```
' 更新適用重要度設定(一括): prtUpdatePriorityAll.vbs
Option Explicit

Dim objWMIService, objShare
Dim objInParam, objOutParams
Dim objPriorityInParam, objPriorityOutParams
Dim objDevClassSet, objDevClass
Dim objToolClassSet, objToolClass
Dim dicDeviceToolNameList, dicDeviceToolName
Dim objDevInfo, objExtendedInfo, intExtendedPos, strExtendedInfo
Dim Count, ErrCount, intReleaseNum, aryReleaseList
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

(1)   ' 操作権限取得プロバイダ名保持リストを作成
set dicDeviceToolNameList = CreateObject("Scripting.Dictionary")
' root/cimv2に接続
Set objWMIService = GetObject("winmgmts:¥¥" & strComputer & "\root\cimv2")

1.    ' 全デバイス用 CIM Provider クラスを取得
Set objDevClassSet = objWMIService.ExecQuery("Select * From meta_class Where __SuperClass = 'HITACHI_DevService'")
' 全ツール用 CIM Provider クラスを取得
Set objToolClassSet = objWMIService.ExecQuery("Select * From meta_class Where __SuperClass = 'HITACHI_ToolService'")

2.    ' 操作権限取得 (デバイス用 CIM Provider)
For Each objDevClass In objDevClassSet
    ' デバイス用 CIM Provider の操作権限を取得
    Call GetOperatingAuthority(objDevClass.Path_.Class)
    ' メソッド実行結果が正常の場合
    If objOutParams.ReturnValue = &H0 Then
        ' デバイス用 CIM Provider クラス名をリストに保持
        dicDeviceToolNameList.Add objDevClass.Path_.Class, objDevClass.Path_.Class
    ' メソッド実行結果が異常の場合
    Else
        ' メソッド実行結果を表示
        Wscript.echo "ReturnValue GetOperatingAuthority (" & objDevClass.Path_.Class & "): " &
        Hex(objOutParams.ReturnValue)
        ' 操作権限取得済のデバイス用 CIM Provider 数とクラス名をリストから取得
        intReleaseNum = dicDeviceToolNameList.Count
        aryReleaseList = dicDeviceToolNameList.Items
        ' 操作権限取得済のデバイス用 CIM Provider の操作権限を解放
        For Count = intReleaseNum - 1 To 0 Step -1
            ' デバイス用 CIM Provider の操作権限を解放
            Call ReleaseOperatingAuthority(aryReleaseList(Count))
            ' メソッド実行結果を表示
            Wscript.echo "ReturnValue ReleaseOperatingAuthority (" & aryReleaseList(Count) & "): " &
            Hex(objOutParams.ReturnValue)
        Next
        ' 処理終了
        Set objExtendedInfo = Nothing
        Set objDevInfo = Nothing
        Set dicDeviceToolNameList = Nothing
        Set objToolClass = Nothing
        Set objToolClassSet = Nothing
        Set objDevClass = Nothing
        Set objDevClassSet = Nothing
```

```

Set objPriorityOutParams = Nothing
Set objPriorityInParam = Nothing
Set objOutParams = Nothing
Set objInParam = Nothing
Set objShare = Nothing
Set objWMIService = Nothing
WScript.Quit
End If
Next

' 操作権限取得 (ツール用 CIM Provider)
For Each objToolClass In objToolClassSet
  ' ツール用 CIM Provider の操作権限を取得
  Call GetOperatingAuthority(objToolClass.Path_.Class)
  ' メソッド実行結果が正常の場合
  If objOutParams.ReturnValue = &H0 Then
    ' ツール用 CIM Provider クラス名をリストに保持
    dicDeviceToolNameList.Add objToolClass.Path_.Class, objToolClass.Path_.Class
  ' メソッド実行結果が異常の場合
  Else
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue GetOperatingAuthority (" & objToolClass.Path_.Class & "): " &
    Hex(objOutParams.ReturnValue)
    ' 操作権限取得済のデバイス・ツール用 CIM Provider 数とクラス名をリストから取得
    intReleaseNum = dicDeviceToolNameList.Count
    aryReleaseList = dicDeviceToolNameList.Items
    ' 操作権限取得済のデバイス・ツール用 CIM Provider の操作権限を解放
    For Count = intReleaseNum - 1 To 0 Step -1
      ' デバイス・ツール用 CIM Provider の操作権限を解放
      Call ReleaseOperatingAuthority(aryReleaseList(Count))
      ' メソッド実行結果を表示
      Wscript.echo "ReturnValue ReleaseOperatingAuthority (" & aryReleaseList(Count) & "): " &
      Hex(objOutParams.ReturnValue)
    Next
    ' 処理終了
    Set objExtendedInfo = Nothing
    Set objDeviceInfo = Nothing
    Set dicDeviceToolNameList = Nothing
    Set objToolClass = Nothing
    Set objToolClassSet = Nothing
    Set objDevClass = Nothing
    Set objDevClassSet = Nothing
    Set objPriorityOutParams = Nothing
    Set objPriorityInParam = Nothing
    Set objOutParams = Nothing
    Set objInParam = Nothing
    Set objShare = Nothing
    Set objWMIService = Nothing
    WScript.Quit
  End If
Next

(2) ' 更新適用重要度設定 (デバイス用 CIM Provider)
For Each objDevClass In objDevClassSet
  ' 設定失敗回数を初期化
  ErrCount = 0
  ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
  Set objOutParams = objWMIService.ExecMethod(objDevClass.Path_.Class, "GetProviderManageInfo")
  ' メソッド実行結果が正常の場合
  If objOutParams.ReturnValue = &H0 Then
    ' プロバイダ管理情報の接続デバイス情報が存在する場合
    If IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
      ' 接続デバイス情報を取得
      For Each objDeviceInfo In objOutParams.DevProviderManageInfo.DeviceInfo
        ' 拡張情報を初期化
        strExtendedInfo = ""
    End If
  End If
Next

```

```

' 拡張情報が存在する場合
If IsNull(objDevInfo.ExtendedInfo) = False Then
    ' 拡張情報を取得
    For Each objExtendedInfo In objDevInfo.ExtendedInfo
        ' 拡張情報の切り出し位置を取得
        intExtendedPos = InStr(objExtendedInfo, "?")
        ' 拡張情報の切り出しが可能な場合
        If intExtendedPos <> 0 Then
            ' 拡張情報を取得
            strExtendedInfo = Mid(objExtendedInfo, 1, intExtendedPos - 1) & ":" &
            Mid(objExtendedInfo, intExtendedPos + 1)
            Exit For
        End If
    Next
End If
' デバイス用 CIM Provider の firm 更新適用重要度を設定
Call SetUpdatePriorityDevice(objDevClass.Path_.Class, "firm")
' メソッド実行結果が異常の場合
If Hex(objPriorityOutParams.ReturnValue) <> &H0 Then
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue SetUpdatePriority (" & objDevClass.Path_.Class & ") " &
    strExtendedInfo & " firm: " & Hex(objPriorityOutParams.ReturnValue)
    ErrCount = ErrCount + 1
End If
' デバイス用 CIM Provider の driver 更新適用重要度を設定
Call SetUpdatePriorityDevice(objDevClass.Path_.Class, "driver")
' メソッド実行結果が異常の場合
If Hex(objPriorityOutParams.ReturnValue) <> &H0 Then
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue SetUpdatePriority (" & objDevClass.Path_.Class & ") " &
    strExtendedInfo & " driver: " & Hex(objPriorityOutParams.ReturnValue)
    ErrCount = ErrCount + 1
End If
' デバイス用 CIM Provider の util 更新適用重要度を設定
Call SetUpdatePriorityDevice(objDevClass.Path_.Class, "util")
' メソッド実行結果が異常の場合
If Hex(objPriorityOutParams.ReturnValue) <> &H0 Then
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue SetUpdatePriority (" & objDevClass.Path_.Class & ") " &
    strExtendedInfo & " util: " & Hex(objPriorityOutParams.ReturnValue)
    ErrCount = ErrCount + 1
End If
' 全モジュールの更新適用重要度メソッド実行結果が正常の場合
ElseIf ErrCount = 0 Then
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue SetUpdatePriority (" & objDevClass.Path_.Class & ") " &
    strExtendedInfo & " : " & Hex(objPriorityOutParams.ReturnValue)
    ErrCount = ErrCount + 1
End If
Next
End If
' メソッド実行結果が異常の場合
Else
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue GetProviderManageInfo (" & objDevClass.Path_.Class & ") : " &
    Hex(objOutParams.ReturnValue)
End If
Next
' 更新適用重要度設定 (ツール用 CIM Provider)
For Each objToolClass In objToolClassSet
    ' ツール用 CIM Provider の util 更新適用重要度を設定
    Call SetUpdatePriorityTool(objToolClass.Path_.Class)
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue SetUpdatePriority (" & objToolClass.Path_.Class & ") : " &
    Hex(objPriorityOutParams.ReturnValue)
Next

```

```

(4)   ' 操作権限解放
      ' 操作権限取得済のデバイス・ツール用 CIM Provider 数とクラス名をリストから取得
      intReleaseNum = dicDeviceToolNameList.Count
      aryReleaseList = dicDeviceToolNameList.Items
      ' 操作権限取得済のデバイス・ツール用 CIM Provider の操作権限を解放
      For Count = intReleaseNum - 1 To 0 Step -1
          ' デバイス・ツール用 CIM Provider の操作権限を解放
          Call ReleaseOperatingAuthority(aryReleaseList(Count))
          ' メソッド実行結果が異常の場合
          If objOutParams.ReturnValue <> &H0 Then
              ' メソッド実行結果を表示
              Wscript.echo "ReturnValue ReleaseOperatingAuthority (" & aryReleaseList(Count) & ")：" &
              Hex(objOutParams.ReturnValue)
              End If
          Next

          Set objExtendedInfo = Nothing
          Set objDevInfo = Nothing
          Set dicDeviceToolNameList = Nothing
          Set objToolClass = Nothing
          Set objToolClassSet = Nothing
          Set objDevClass = Nothing
          Set objDevClassSet = Nothing
          Set objPriorityOutParams = Nothing
          Set objPriorityInParam = Nothing
          Set objOutParams = Nothing
          Set objInParam = Nothing
          Set objShare = Nothing
          Set objWMIService = Nothing

          ' /////////////////////////////////
          ' 操作権限取得
          ' ///////////////////////////////
Sub GetOperatingAuthority(DeviceToolName)

    ' デバイス・ツール用 CIM Provider クラスを取得
    Set objShare = objWMIService.Get(DeviceToolName)
    ' 操作権限取得メソッドのパラメータセットを取得
    Set objInParam = objShare.Methods_("GetOperatingAuthority").inParameters.SpawnInstance_()
    ' パラメータセットにパラメータを設定
    ' アプリケーションキー "CLI"
    objInParam.Properties_.Item("AppKey") = "CLI"

    ' 操作権限取得メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod(DeviceToolName, "GetOperatingAuthority", objInParam)
End Sub

          ' /////////////////////////////////
          ' 操作権限解放
          ' ///////////////////////////////
Sub ReleaseOperatingAuthority(DeviceToolName)

    ' デバイス・ツール用 CIM Provider クラスを取得
    Set objShare = objWMIService.Get(DeviceToolName)
    ' 操作権限解放メソッドのパラメータセットを取得
    Set objInParam = objShare.Methods_("ReleaseOperatingAuthority").inParameters.SpawnInstance_()
    ' パラメータセットにパラメータを設定
    ' アプリケーションキー "CLI"
    objInParam.Properties_.Item("AppKey") = "CLI"

    ' 操作権限解放メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod(DeviceToolName, "ReleaseOperatingAuthority", objInParam)
End Sub

```

```

' //////////////////////////////////////////////////////////////////
' 更新適用重要度設定 (デバイス用 CIM Provider)
' //////////////////////////////////////////////////////////////////
Sub SetUpdatePriorityDevice(DeviceName, ModuleKind)

    ' デバイス用 CIM Provider クラスを取得
    Set objShare = objWMIService.Get(DeviceName)
    ' 更新適用重要度設定 (デバイス用 CIM Provider) メソッドのパラメータセットを取得
    Set objPriorityInParam = objShare.Methods_("SetUpdatePriority").inParameters.SpawnInstance_()
    ' パラメータセットにパラメータを設定
    ' アプリケーションキー "CLI"
    objPriorityInParam.Properties_.Item("AppKey") = "CLI"
    ' 重要度
    objPriorityInParam.Properties_.Item("UpdatePriority") = 1
    ' デバイス ID
    objPriorityInParam.Properties_.Item("DeviceID") = objDeviceInfo.DeviceID
    ' モジュール種別
    objPriorityInParam.Properties_.Item("ModuleKind") = ModuleKind

    ' 更新適用重要度設定 (デバイス用 CIM Provider) メソッドを実行して戻り値を取得
    Set objPriorityOutParams = objWMIService.ExecMethod(DeviceName, "SetUpdatePriority", objPriorityInParam)
End Sub

' //////////////////////////////////////////////////////////////////
' 更新適用重要度設定 (ツール用 CIM Provider)
' //////////////////////////////////////////////////////////////////
Sub SetUpdatePriorityTool(ToolName)

    ' ツール用 CIM Provider クラスを取得
    Set objShare = objWMIService.Get(ToolName)
    ' 更新適用重要度設定 (ツール用 CIM Provider) メソッドのパラメータセットを取得
    Set objPriorityInParam = objShare.Methods_("SetUpdatePriority").inParameters.SpawnInstance_()
    ' パラメータセットにパラメータを設定
    ' アプリケーションキー "CLI"
    objPriorityInParam.Properties_.Item("AppKey") = "CLI"
    ' 重要度
    objPriorityInParam.Properties_.Item("UpdatePriority") = 1

    ' 更新適用重要度設定 (ツール用 CIM Provider) メソッドを実行して戻り値を取得
    Set objPriorityOutParams = objWMIService.ExecMethod(ToolName, "SetUpdatePriority", objPriorityInParam)
End Sub

```

更新適用重要度設定のサンプルソース 2

```
' 更新適用重要度設定(個別): prtUpdatePriority.vbs
Option Explicit

Dim objWMIService, objShare
Dim objInParam, objOutParams
Dim objPriorityInParam, objPriorityOutParams
Dim objDeviceInfo, objExtendedInfo, intExtendedPos, strExtendedInfo
Dim ErrCount
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\CIMV2")

(1) ' 操作権限取得
    ' RAID用CIM Providerの操作権限を取得
    Set objShare = objWMIService.Get("HITACHI_DevServiceRaid")
    ' 操作権限取得メソッドのパラメータセットを取得
    Set objInParam = objShare.Methods_("GetOperatingAuthority").inParameters.SpawnInstance_()
    ' パラメータセットにパラメータを設定
    ' アプリケーションキー "CLI"
    objInParam.Properties_.Item("AppKey") = "CLI"
    ' 操作権限取得メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod("HITACHI_DevServiceRaid", "GetOperatingAuthority", objInParam)
    ' メソッド実行結果が異常の場合
    If objOutParams.ReturnValue <> &H0 Then
        ' メソッド実行結果を表示
        Wscript.echo "ReturnValue GetOperatingAuthority (Raid): " & Hex(objOutParams.ReturnValue)
        ' RAID用CIM Providerの操作権限を解放
        Call ReleaseOperatingAuthority("HITACHI_DevServiceRaid")
        ' メソッド実行結果を表示
        Wscript.echo "ReturnValue ReleaseOperatingAuthority (Raid): " & Hex(objOutParams.ReturnValue)
        ' 処理終了
        Set objExtendedInfo = Nothing
        Set objDeviceInfo = Nothing
        Set objPriorityOutParams = Nothing
        Set objPriorityInParam = Nothing
        Set objOutParams = Nothing
        Set objInParam = Nothing
        Set objShare = Nothing
        Set objWMIService = Nothing
        WScript.Quit
    End If

(2) ' 更新適用重要度設定
    ' 設定失敗回数を初期化
    ErrCount = 0
    1. ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
        Set objOutParams = objWMIService.ExecMethod("HITACHI_DevServiceRaid", "GetProviderManageInfo")
    2. ' メソッド実行結果が正常の場合
        If objOutParams.ReturnValue = &H0 Then
            ' プロバイダ管理情報の接続デバイス情報が存在する場合
            If IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
                ' 接続デバイス情報を取得
                For Each objDeviceInfo In objOutParams.DevProviderManageInfo.DeviceInfo
                    ' 拡張情報を初期化
                    strExtendedInfo = ""
            End If
        End If
    End If
```

```

        ' 拡張情報が存在する場合
        If IsNull(objDevInfo.ExtendedInfo) = False Then
            ' 拡張情報を取得
            For Each objExtendedInfo In objDevInfo.ExtendedInfo
                ' 拡張情報の切り出し位置を取得
                intExtendedPos = InStr(objExtendedInfo, "?")
                ' 拡張情報の切り出しが可能な場合
                If intExtendedPos <> 0 Then
                    ' 拡張情報を取得
                    strExtendedInfo = Mid(objExtendedInfo, 1, intExtendedPos - 1) & ":" &
                    Mid(objExtendedInfo, intExtendedPos + 1)
                    Exit For
                End If
            Next
        End If
    End If
    ' RAID 用 CIM Provider の firm 更新適用重要度を設定
    Call SetUpdatePriority("HITACHI_DevServiceRaid", "firm")
    ' メソッド実行結果が異常の場合
    If Hex(objPriorityOutParams.ReturnValue) <> &H0 Then
        ' メソッド実行結果を表示
        Wscript.echo "ReturnValue SetUpdatePriority (Raid) " & strExtendedInfo & " firm: " &
        Hex(objPriorityOutParams.ReturnValue)
        ErrCount = ErrCount + 1
    End If
    ' RAID 用 CIM Provider の driver 更新適用重要度を設定
    Call SetUpdatePriority("HITACHI_DevServiceRaid", "driver")
    ' メソッド実行結果が異常の場合
    If Hex(objPriorityOutParams.ReturnValue) <> &H0 Then
        ' メソッド実行結果を表示
        Wscript.echo "ReturnValue SetUpdatePriority (Raid) " & strExtendedInfo & " driver: " &
        Hex(objPriorityOutParams.ReturnValue)
        ErrCount = ErrCount + 1
    End If
    ' RAID 用 CIM Provider の util 更新適用重要度を設定
    Call SetUpdatePriority("HITACHI_DevServiceRaid", "util")
    ' メソッド実行結果が異常の場合
    If Hex(objPriorityOutParams.ReturnValue) <> &H0 Then
        ' メソッド実行結果を表示
        Wscript.echo "ReturnValue SetUpdatePriority (Raid) " & strExtendedInfo & " util: " &
        Hex(objPriorityOutParams.ReturnValue)
        ErrCount = ErrCount + 1
    End If
    ' 全モジュールの更新適用重要度メソッド実行結果が正常の場合
    ElseIf ErrCount = 0 Then
        ' メソッド実行結果を表示
        Wscript.echo "ReturnValue SetUpdatePriority (Raid) " & strExtendedInfo & " :" &
        Hex(objPriorityOutParams.ReturnValue)
        ErrCount = ErrCount + 1
    End If
    Next
End If
' メソッド実行結果が異常の場合
Else
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue GetProviderManageInfo (Raid) : " & Hex(objOutParams.ReturnValue)
End If

(3)
1.      ' 操作権限解放
        ' RAID 用 CIM Provider の操作権限を解放
        Call ReleaseOperatingAuthority("HITACHI_DevServiceRaid")
        ' メソッド実行結果が異常の場合
        If objOutParams.ReturnValue <> &H0 Then
            ' メソッド実行結果を表示
            Wscript.echo "ReturnValue ReleaseOperatingAuthority (Raid): " & Hex(objOutParams.ReturnValue)
        End If

        Set objExtendedInfo = Nothing

```

```

Set objDevInfo = Nothing
Set objPriorityOutParams = Nothing
Set objPriorityInParam = Nothing
Set objOutParams = Nothing
Set objInParam = Nothing
Set objShare = Nothing
Set objWMIService = Nothing

' /////////////////////////////////////////////////
' 操作権限取得
' /////////////////////////////////////////////////
Sub GetOperatingAuthority(DeviceName)

    ' デバイス用 CIM Provider クラスを取得
    Set objShare = objWMIService.Get(DeviceName)
    ' 操作権限取得メソッドのパラメータセットを取得
    Set objInParam = objShare.Methods_("GetOperatingAuthority").inParameters.SpawnInstance_()
    ' パラメータセットにパラメータを設定
    ' アプリケーションキー "CLI"
    objInParam.Properties_.Item("AppKey") = "CLI"

    ' 操作権限取得メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod(DeviceName, "GetOperatingAuthority", objInParam)
End Sub

' /////////////////////////////////////////////////
' 操作権限解放
' /////////////////////////////////////////////////
Sub ReleaseOperatingAuthority(DeviceToolName)

    ' デバイス・ツール用 CIM Provider クラスを取得
    Set objShare = objWMIService.Get(DeviceToolName)
    ' 操作権限解放メソッドのパラメータセットを取得
    Set objInParam = objShare.Methods_("ReleaseOperatingAuthority").inParameters.SpawnInstance_()
    ' パラメータセットにパラメータを設定
    ' アプリケーションキー "CLI"
    objInParam.Properties_.Item("AppKey") = "CLI"

    ' 操作権限解放メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod(DeviceToolName, "ReleaseOperatingAuthority", objInParam)
End Sub

' /////////////////////////////////////////////////
' 更新適用重要度設定
' /////////////////////////////////////////////////
Sub SetUpdatePriority(DeviceName, ModuleKind)

    ' CIM Provider クラスを取得
    Set objShare = objWMIService.Get(DeviceName)
    ' 更新適用重要度設定 (デバイス用 CIM Provider) メソッドのパラメータセットを取得
    Set objPriorityInParam = objShare.Methods_("SetUpdatePriority").inParameters.SpawnInstance_()
    ' パラメータセットにパラメータを設定
    ' アプリケーションキー "CLI"
    objPriorityInParam.Properties_.Item("AppKey") = "CLI"
    ' 重要度
    objPriorityInParam.Properties_.Item("UpdatePriority") = 1
    ' デバイス ID
    objPriorityInParam.Properties_.Item("DeviceID") = objDevInfo.DeviceID
    ' モジュール種別
    objPriorityInParam.Properties_.Item("ModuleKind") = ModuleKind

    ' 更新適用重要度設定 (デバイス用 CIM Provider) メソッドを実行して戻り値を取得
    Set objPriorityOutParams = objWMIService.ExecMethod(DeviceName, "SetUpdatePriority", objPriorityInParam)
End Sub

```

アップデート予定リスト取得(CLI)

概要

GUI では、以下の処理を実行します。

1. アップデート予定リスト取得

デバイス・ツール用 CIM Provider のアップデート予定リスト取得メソッドを実行し、アップデート予定リストを取得して表示します。

このとき、アップデート予定リストのうち、更新対象外に設定されているデバイスの情報は表示しません。

サンプルソースの処理内容詳細

サンプルソースでは、ロールバック指定を False に設定してデバイス用 CIM Provider 及びツール用 CIM Provider のアップデート予定リストを取得します。

また、更新可否に関わらず全てのアップデート予定をデバイスリストのようにツリー表示を行います。

更新可否が更新対象外のアップデート予定は、"No Select Modules" と表示します。

(1) システムデバイスアップデート予定リスト取得

1. ロールバック指定を False に設定して BIOS/EFI 用 CIM Provider のアップデート予定リスト取得メソッドを実行し、アップデート予定リストを取得する

2. メソッド実行結果が正常かつアップデート予定リストが存在する場合、アップデート予定の更新可否を取得する

メソッド実行結果が異常の場合、メソッド実行結果を表示して BMC 用 CIM Provider のアップデート予定リストを取得する

3. システムデバイス名(固定文言: System)をツリーの頂点ノードに表示する

4. BIOS/EFI デバイスをシステムデバイスとして、デバイス名称を固定文言(BIOS/EFI デバイス: BIOS/EFI)で固定ツリー表示する

5. アップデート予定を表示する

6. メソッド実行結果を表示する

7. ロールバック指定を False に設定して BMC 用 CIM Provider のアップデート予定リスト取得メソッドを実行し、アップデート予定リストを取得する

8. メソッド実行結果が正常かつアップデート予定リストが存在する場合、アップデート予定の更新可否を取得する

メソッド実行結果が異常の場合、メソッド実行結果を表示して次のデバイス用 CIM Provider のアップデート予定リストを取得する

9. BMC デバイスをシステムデバイスとして、デバイス名称を固定文言(BMC デバイス: BMC)で固定ツリー表示する

10. アップデート予定を表示する
 11. メソッド実行結果を表示する
- (2) デバイスアップデート予定リスト取得
1. WMI のクエリを実行して全デバイス用 CIM Provider クラスを取得する
 2. ロールバック指定を False に設定して BIOS/EFI と BMC 以外の全デバイス用 CIM Provider のアップデート予定リスト取得メソッドを実行し、アップデート予定リストを取得する
 3. メソッド実行結果が正常かつアップデート予定リストが存在する場合、プロバイダ管理情報を取得する
メソッド実行結果が異常の場合、メソッド実行結果を表示して次のデバイス用 CIM Provider のアップデート予定リストを取得する
 4. BIOS/EFI と BMC 以外の全デバイス用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する
 5. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する
メソッド実行結果が異常の場合、メソッド実行結果を表示して次のデバイス用 CIM Provider のアップデート予定リストを取得する
 6. WMI のクエリを実行して、PCI デバイス情報のデバイス名称(詳細)とデバイス ID を取得する
 7. プロバイダ名をツリーの頂点ノードに表示する
 8. アップデート予定のデバイス ID と接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合、PCI デバイス情報のデバイス名称(詳細)をツリー表示する
 9. 接続デバイス情報の拡張情報をキーとバリューに分割して表示する
拡張情報がキーとバリューに分割できない場合は、拡張情報を表示しない
 10. アップデート予定を表示する
 11. メソッド実行結果を表示する
- (3) ツールアップデート予定リスト取得
1. WMI のクエリを実行して全ツール用 CIM Provider クラスを取得する
 2. 全ツール用 CIM Provider のアップデート予定リスト取得メソッドを実行し、アップデート予定リストを取得する
 3. メソッド実行結果が正常かつアップデート予定リストが存在する場合、アップデート予定の更新可否を取得する
メソッド実行結果が異常の場合、メソッド実行結果を表示して次のツール用 CIM Provider のアップデート予定リストを取得する
 4. ツールリスト名(固定文言: Tool)をツリーの頂点ノードに表示し、プロバイダ名をツリー表示する

5. アップデート予定を表示する
6. メソッド実行結果を表示する



- ツールのロールバックについて

Hitachi Server Navigator では、ツールはアップデートのみ行い、ロールバックは行いません。

ツール用 CIM Provider のアップデート予定リスト取得メソッドで取得できるアップデート予定リストは、アップデート時の予定リストとなります。

アップデート予定リスト取得のサンプルソース

```

' アップデート予定リスト取得: prtUpdatePlans.vbs
Option Explicit

Dim objWMIService, objShare
Dim objInParam, objOutParams
Dim objDevOutParams
Dim objClassSet, objClass
Dim objPnPSet, objPnP
Dim objUpdatePlanInfo, objDeviceInfo, objExtendedInfo
Dim strUpdatePlan, intExtendedPos
Dim ProviderList(1), strProvider
Dim strComputer
Const BiosProvider = "HITACHI_DevServiceBios"
Const BMCPProvider = "HITACHI_DevServiceBMC"

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

(1)   ' システムデバイスアップデート予定リスト取得
      ' システムデバイスプロバイダリストを作成
ProviderList(0) = BiosProvider
ProviderList(1) = BMCPProvider
      ' root/cimv2に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\cimv2")

3.    ' システムデバイスツリー表示
Wscript.echo "System"
For Each strProvider In ProviderList
    ' BIOS/EFI用CIM Providerの場合
    If strProvider = BiosProvider Then
        ' BIOS/EFIデバイスツリー表示
        Wscript.echo " BIOS/EFI"
    ' BMC用CIM Providerの場合
    ElseIf strProvider = BMCPProvider Then
        ' BMCデバイスツリー表示
        Wscript.echo " BMC"
    End If
    ' デバイス用CIM Providerクラスを取得
    Set objShare = objWMIService.Get(strProvider)
    ' アップデート予定リスト取得メソッドのパラメータセットを取得
    Set objInParam = objShare.Methods_("GetUpdatePlans").inParameters.SpawnInstance_()
    ' パラメータセットにパラメータを設定
    ' ロールバック指定 True:ロールバック, False:アップデート
    objInParam.Properties_.Item("Rollback") = False
    ' アップデート予定リスト取得メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod(strProvider, "GetUpdatePlans")
    ' メソッド実行結果が正常の場合
    If objOutParams.ReturnValue = &H0 Then
        ' アップデート予定リストが存在する場合
        If IsNull(objOutParams.UpdatePlansList) = False Then
            ' アップデート予定を取得
            For Each objUpdatePlanInfo In objOutParams.UpdatePlansList
                ' システムデバイスアップデート予定を表示
                Call DisplayUpdatePlan()
            Next
        End If
        ' メソッド実行結果を表示
        Wscript.echo " ReturnValue GetUpdatePlans (" & Replace(strProvider, "HITACHI_DevService", "") & ")"
    " & Hex(objOutParams.ReturnValue)
        ' メソッド実行結果が異常の場合
    End If
6. 11.
!
```

```

        Else
            ' メソッド実行結果を表示
            Wscript.echo " ReturnValue GetUpdatePlans (" & Replace(strProvider, "HITACHI_DevService", "") & ")
" & Hex(objOutParams.ReturnValue)
        End If
    Next

    ' デバイスアップデート予定リスト取得
    ' 全デバイス用 CIM Provider クラスを取得
    Set objClassSet = objWMIService.ExecQuery("Select * From meta_class Where _SuperClass ='HITACHI_DevService'")

    For Each objClass In objClassSet
        ' BIOS/EFI 用 CIM Provider でも BMC 用 CIM Provider でもない場合
        If objClass.Path_.Class <> BiosProvider And objClass.Path_.Class <> BMCProvider Then
            ' デバイス用 CIM Provider クラスを取得
            Set objShare = objWMIService.Get(objClass.Path_.Class)
            ' アップデート予定リスト取得メソッドのパラメータセットを取得
            Set objInParam = objShare.Methods_("GetUpdatePlans").inParameters.SpawnInstance_()
            ' パラメータセットにパラメータを設定
            ' ロールバック指定 True:ロールバック, False:アップデート
            objInParam.Properties_.Item("Rollback") = False
            ' アップデート予定リスト取得メソッドを実行して戻り値を取得
            Set objOutParams = objWMIService.ExecMethod(objClass.Path_.Class, "GetUpdatePlans")
            ' メソッド実行結果が正常の場合
            If objOutParams.ReturnValue = &H0 Then
                ' アップデート予定リストが存在する場合
                If IsNull(objOutParams.UpdatePlansList) = False Then
                    ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
                    Set objDevOutParams = objWMIService.ExecMethod(objClass.Path_.Class, "GetProviderManageInfo")
                    ' メソッド実行結果が正常の場合
                    If objDevOutParams.ReturnValue = &H0 Then
                        ' プロバイダ管理情報の接続デバイス情報が存在する場合
                        If IsNull(objDevOutParams.DevProviderManageInfo.DeviceInfo) = False Then
                            ' 接続デバイス情報を取得
                            For Each objDevInfo In objDevOutParams.DevProviderManageInfo.DeviceInfo
                                ' OS から PCI デバイス情報を取得
                                Set objPnPSet = objWMIService.ExecQuery_(
"SELECT Caption,DeviceID FROM Win32_PnPEntity WHERE DeviceID LIKE 'PCI%' AND
ConfigManagerErrorCode = '0' AND Status = 'OK'")
                                ' プロバイダ名を表示
                                Wscript.Echo objDevOutParams.DevProviderManageInfo.ProviderName
                                ' 接続デバイス情報と合致する PCI デバイス情報を検索
                                For Each objPnP In objPnPSet
                                    ' 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合
                                    If objDevInfo.DeviceID = objPnP.DeviceID Then
                                        ' デバイス名称をツリー表示
                                        Wscript.Echo " " & objPnP.Caption
                                        ' 拡張情報が存在する場合
                                        If IsNull(objDevInfo.ExtendedInfo) = False Then
                                            ' 拡張情報を取得
                                            For Each objExtendedInfo In objDevInfo.ExtendedInfo
                                                ' 拡張情報の切り出し位置を取得
                                                intExtendedPos = InStr(objExtendedInfo, "?")
                                                ' 拡張情報の切り出しが可能な場合
                                                If intExtendedPos <> 0 Then
                                                    ' 拡張情報を表示
                                                    Wscript.Echo " " & Mid(objExtendedInfo, 1, intExtendedPos
- 1) & ":" & Mid(objExtendedInfo, intExtendedPos + 1)
                                                End If
                                            Next
                                        End If
                                        ' アップデート予定を取得
                                        For Each objUpdatePlanInfo In objOutParams.UpdatePlansList
                                            ' PCI デバイス情報のデバイス ID とアップデート予定のデバイス ID が一致
                                            If objPnP.DeviceID = objUpdatePlanInfo.DeviceID Then

```

```

        ' デバイスアップデート予定をツリー表示
        Call DisplayUpdatePlan()
    End If
    Next
    Exit For
End If
Next
Next
' メソッド実行結果を表示
Wscript.echo "ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
"HTACHI_DevService", "") & "): " & Hex(objOutParams.ReturnValue)
' メソッド実行結果を表示
Wscript.echo "ReturnValue GetUpdatePlans (" & Replace(objClass.Path_.Class,
"HTACHI_DevService", "") & "): " & Hex(objOutParams.ReturnValue)
End If
' メソッド実行結果が異常の場合
Else
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue GetProviderManageInfo (" & Replace(objClass.Path_.Class,
"HTACHI_DevService", "") & "): " & Hex(objOutParams.ReturnValue)
End If
End If
' メソッド実行結果が異常の場合
Else
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue GetUpdatePlans (" & Replace(objClass.Path_.Class, "HTACHI_DevService",
 "") & "): " & Hex(objOutParams.ReturnValue)
End If
End If
Next

(3)
1. ' ツールアップデート情報表示
' 全ツール用 CIM Provider クラスを取得
Set objClassSet = objWMIService.ExecQuery("Select * From meta_class Where __SuperClass='HTACHI_ToolService'")

2. ' ツールツリー表示
Wscript.echo "Tool"
For Each objClass In objClassSet
    ' アップデート予定リスト取得メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod(objClass.Path_.Class, "GetUpdatePlans")
    ' メソッド実行結果が正常の場合
    If objOutParams.ReturnValue = &H0 Then
        ' アップデート予定リストが存在する場合
        If IsNull(objOutParams.UpdatePlansList) = False Then
            For Each objUpdatePlanInfo In objOutParams.UpdatePlansList
                ' プロバイダ名を表示
                Wscript.Echo " " & objUpdatePlanInfo.ProviderName
                Exit For
            Next
            For Each objUpdatePlanInfo In objOutParams.UpdatePlansList
                ' ツールアップデート予定をツリー表示
                Call DisplayUpdatePlan()
            Next
        End If
        ' メソッド実行結果が異常の場合
        Else
            ' メソッド実行結果を表示
            Wscript.echo " ReturnValue GetUpdatePlans (" & Replace(objClass.Path_.Class,
"HTACHI_ToolService", "") & "): " & Hex(objOutParams.ReturnValue)
        End If
    End If
    ' メソッド実行結果が異常の場合
    Else
        ' メソッド実行結果を表示
        Wscript.echo " ReturnValue GetUpdatePlans (" & Replace(objClass.Path_.Class, "HTACHI_ToolService",
 "") & "): " & Hex(objOutParams.ReturnValue)
    End If
Next

```

```

Set objExtendedInfo = Nothing
Set objDevInfo = Nothing
Set objUpdatePlanInfo = Nothing
Set objPnP = Nothing
Set objPnPSet = Nothing
Set objClass = Nothing
Set objClassSet = Nothing
Set objDevOutParams = Nothing
Set objOutParams = Nothing
Set objInParam = Nothing
Set objShare = Nothing
Set objWMIService = Nothing

' /////////////////////////////////////////////////
' アップデート予定表示
' /////////////////////////////////////////////////
Sub DisplayUpdatePlan()

    ' 更新可否が更新非対象でないアップデート予定の場合
    If objUpdatePlanInfo.UpdateJudgment <> 0 Then
        ' モジュール種別
        If objUpdatePlanInfo.ModuleKind <> "" Then
            strUpdatePlan = objUpdatePlanInfo.ModuleKind
        Else
            strUpdatePlan = "-"
        End If
        ' 稼動バージョン
        If objUpdatePlanInfo.CurrentVendorVersion <> "" Then
            strUpdatePlan = strUpdatePlan & " " & objUpdatePlanInfo.CurrentVendorVersion
        Else
            strUpdatePlan = strUpdatePlan & " -"
        End If
        ' 更新バージョン ロールバック時は旧稼動バージョン
        If objUpdatePlanInfo.NextVendorVersion <> "" Then
            strUpdatePlan = strUpdatePlan & " " & objUpdatePlanInfo.NextVendorVersion
        Else
            strUpdatePlan = strUpdatePlan & " -"
        End If
        ' 更新可否
        strUpdatePlan = strUpdatePlan & " " & ConvertUpdateJudgment(objUpdatePlanInfo.UpdateJudgment)
        ' 更新可否が更新非対象のアップデート予定の場合
    Else
        ' 更新可否 (更新非対象)
        strUpdatePlan = ConvertUpdateJudgment(objUpdatePlanInfo.UpdateJudgment)
    End If
    ' アップデート予定を表示
    Wscript.echo " " & strUpdatePlan
End Sub

' /////////////////////////////////////////////////
' 更新可否変換
' /////////////////////////////////////////////////
Function ConvertUpdateJudgment(Judgment)

    Select Case Judgment
        Case 0
            ' 更新非対象
            ConvertUpdateJudgment = "No Select Modules"
        Case 1
            ' 更新ありアップデート可能
            ConvertUpdateJudgment = "YES"
        Case 2
            ' 更新ありアップデート条件 NG
            ConvertUpdateJudgment = "NO (There is a problem in update condition.)"
    End Select
End Function

```

```
Case 3
    , 更新あり依存関係 NG
    ConvertUpdateJudgment = "NO (There is a problem in devendance.)"
Case 4
    , 更新なし (更新モジュールなし)
    ConvertUpdateJudgment = "NO (Next version does not exist.)"
Case Else
    , 更新非対象
    ConvertUpdateJudgment = "No Select Modules"
End Select
End Function
```

アップデート履歴取得(CLI)

概要

GUI では、以下の処理を実行します。

1. アップデート履歴取得

ログ収集 CIM Provider のアップデート履歴収集メソッドを実行し、CIM Provider が出力した稼動ログのうち、アップデート履歴ログを取得して表示します。アップデート履歴ログは、CIM Provider 別に日時の昇順にソートされます。

アップデート履歴ログは、CIM Provider 別に最大 20 行取得します。

サンプルソースの処理内容詳細

サンプルソースでは、GUI と同様の処理を実行します。

(1) アップデート履歴表示

1. ログ収集 Provider のアップデート履歴収集メソッドを実行し、アップデート履歴ログリストを取得する
2. メソッド実行結果が正常かつアップデート履歴ログリストが存在する場合、アップデート履歴ログリストの情報を表示する
3. メソッド実行結果を表示する

アップデート履歴取得のサンプルソース

```
' アップデート履歴取得: prtUpdateHistory.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim objVersionUpdateLog
Dim strDevice, strDateTime, strBeforeVersion, strAfterVersion
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2に接続
Set objWMIService = GetObject("winmgmts::" & strComputer & "\root\CIMV2")

(1)
1.   ' アップデート履歴収集メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_LogManager", "CollectVersionUpLog")
' メソッド実行結果が正常かつアップデート履歴ログリストが存在する場合
If objOutParams.ReturnValue = &H0 And IsNull(objOutParams.VersionUpdateLogList) = False Then
    ' アップデート履歴ログリストを表示
    Wscript.echo "VersionUpdateLog:"
    For Each objVersionUpdateLog In objOutParams.VersionUpdateLogList
        ' デバイス情報
        strDevice = objVersionUpdateLog.ProviderName & " " & objVersionUpdateLog.VendorName & " - " &
        objVersionUpdateLog.UniqueName & " " & objVersionUpdateLog.ModuleKind
        ' 日時
        strDateTime = objVersionUpdateLog.DateTime
        ' アップデート前バージョン
        strBeforeVersion = objVersionUpdateLog.BeforeVersion
        ' アップデート後バージョン
        strAfterVersion = objVersionUpdateLog.AfterVersion
        ' アップデート履歴ログを表示
        Wscript.echo strDevice & " " & strDateTime & " " & strBeforeVersion & " " & strAfterVersion
    Next
End If
' メソッド実行結果を表示
Wscript.echo "ReturnValue CollectVersionUpLog: " & Hex(objOutParams.ReturnValue)

Set objOutParams = Nothing
Set objWMIService = Nothing
```

監視設定取得(CLI)

概要

GUI では、以下の処理を実行します。

1. 監視設定項目取得

監視設定項目を監視 CIM Provider から取得します。

サンプルソースの処理内容詳細

サンプルソースでは、GUI と同様の処理を実行します。

(1) 監視設定項目取得

1. 設定項目名称に"ErrorWatching"、"TimeoutAction"、"TimeoutInterval"をそれぞれ設定して値取得メソッドを実行し、設定値と型を取得する

2. メソッド実行結果が正常の場合、設定値と型を取得する

メソッド実行結果が異常の場合、メソッド実行結果を表示して監視設定項目取得処理を終了する

3. メソッド実行結果を表示する

監視設定取得のサンプルソース

```
' 監視設定取得: prtAliveMonitorSetting.vbs
Option Explicit

Dim objWMIService, objShare
Dim objInParam, objOutParams
Dim ParameterList(2), Parameter
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' 監視設定項目リスト作成
ParameterList(0) = "ErrorWatching"
ParameterList(1) = "TimeoutAction"
ParameterList(2) = "TimeoutInterval"
' root/cimv2に接続
Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\cimv2")

(1)   ' 監視 CIM Provider クラスを取得
Set objShare = objWMIService.Get("HITACHI_ToolServiceWD")
' 監視設定項目取得
1.    For Each Parameter In ParameterList
    ' 値取得メソッドのパラメータセットを取得
    Set objInParam = objShare.Methods_("GetParameter").inParameters.SpawnInstance_()
    ' 監視設定項目 ErrorWatching, TimeoutAction, TimeoutInterval
    objInParam.Properties_.Item("Name") = Parameter
    ' 値取得メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod("HITACHI_ToolServiceWD", "GetParameter", objInParam)
    ' メソッド実行結果が正常の場合
2.    If Hex(objOutParams.ReturnValue) = &H0 Then
        ' 設定値を表示
        Wscript.echo Parameter & " Value: " & objOutParams.Value
        ' 型を取得
        Wscript.echo Parameter & " Type: " & objOutParams.Type
        ' メソッド実行結果を表示
        Wscript.echo "ReturnValue GetParameter " & Parameter & ":" & Hex(objOutParams.ReturnValue)
    ' メソッド実行結果が異常の場合
3.    Else
        ' メソッド実行結果を表示
        Wscript.echo "ReturnValue GetParameter " & Parameter & ":" & Hex(objOutParams.ReturnValue)
        ' 処理終了
        Set objOutParams = Nothing
        Set objInParam = Nothing
        Set objShare = Nothing
        Set objWMIService = Nothing
        WScript.Quit
    End If
Next

Set objOutParams = Nothing
Set objInParam = Nothing
Set objShare = Nothing
Set objWMIService = Nothing
```

監視設定更新(CLI)

概要

GUI では、以下の処理を実行します。

1. 監視設定項目更新

監視設定項目の設定値を設定し、すべての設定値が正常に設定できた場合設定値を更新する。

サンプルソースの処理内容詳細

サンプルソースでは、GUI と同様の処理を実行します。

(1) 監視設定項目設定

1. 設定項目名称に"ErrorWatching"、"TimeoutAction"、"TimeoutInterval"と、設定値に値をそれぞれ設定して値設定メソッドを実行して値を設定する

2. メソッド実行結果が正常の場合、値更新メソッドを実行して値を更新する

メソッド実行結果が異常の場合、メソッド実行結果を表示して値削除メソッドを実行して値の設定を削除し、監視設定項目設定処理を終了する

3. メソッド実行結果を表示する

監視設定更新のサンプルソース

```
' 監視設定更新: setAliveMonitorSetting.vbs
Option Explicit

Dim objWMIService, objShare
Dim objInParam, objOutParams, objClearOutParams
Dim dicParameterList, dicParameter
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' 設定値リストを作成
set dicParameterList = CreateObject("Scripting.Dictionary")
dicParameterList.Add "ErrorWatching", "TRUE"
dicParameterList.Add "TimeoutAction", "NoAction"
dicParameterList.Add "TimeoutInterval", "10"
' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts::" & strComputer & "\root\cimv2")

(1) ' 監視 CIM Provider クラスを取得
Set objShare = objWMIService.Get("HITACHI_ToolServiceWD")

1. ' 値設定
For Each dicParameter In dicParameterList
    ' 値設定メソッドを実行
    Call SetParameter(dicParameter, dicParameterList.Item(dicParameter))
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue SetParameter " & dicParameter & ":" & Hex(objOutParams.ReturnValue)
    ' メソッド実行結果が異常の場合
    If Hex(objOutParams.ReturnValue) <> &H0 Then
        ' 値削除メソッドを実行して戻り値を取得
        Set objClearOutParams = objWMIService.ExecMethod("HITACHI_ToolServiceWD", "ClearParameters")
        ' メソッド実行結果を表示
        Wscript.echo "ReturnValue ClearParameters: " & Hex(objClearOutParams.ReturnValue)
        ' 処理終了
        Set dicParameterList = Nothing
        Set objClearOutParams = Nothing
        Set objOutParams = Nothing
        Set objInParam = Nothing
        Set objShare = Nothing
        Set objWMIService = Nothing
        WScript.Quit
    End If
Next

2. ' 値更新メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_ToolServiceWD", "UpdateParameters")
' メソッド実行結果を表示
Wscript.echo "ReturnValue UpdateParameters: " & Hex(objOutParams.ReturnValue)
' メソッド実行結果が異常の場合
If Hex(objOutParams.ReturnValue) <> &H0 Then
    ' 値削除メソッドを実行して戻り値を取得
    Set objClearOutParams = objWMIService.ExecMethod("HITACHI_ToolServiceWD", "ClearParameters")
    ' メソッド実行結果を表示
    Wscript.echo "ReturnValue ClearParameters: " & Hex(objClearOutParams.ReturnValue)
    ' 処理終了
    Set dicParameterList = Nothing
    Set objClearOutParams = Nothing
    Set objOutParams = Nothing
    Set objInParam = Nothing
```

```

Set objShare = Nothing
Set objWMIService = Nothing
WScript.Quit
End If

Set objOutParams = Nothing
Set objInParam = Nothing
Set objShare = Nothing
Set objWMIService = Nothing

' /////////////////////////////////
' SetParameter
' /////////////////////////////////
Sub SetParameter (name, value)

    ' 値設定メソッドのパラメータセットを取得
    Set objInParam = objShare.Methods_ ("Setparameter"). inParameters. SpawnInstance_()
    ' パラメータセットにパラメータを設定
    ' 設定項目名称
    objInParam.Properties_. Item("Name") = name
    ' 設定値
    objInParam.Properties_. Item("Value") = value

    ' 値設定メソッドを実行して戻り値を取得
    Set objOutParams = objWMIService.ExecMethod("HITACHI_ToolServiceWD", "SetParameter", objInParam)
End Sub

```

連携ツール表示(CLI)

概要

GUI では、以下の処理を実行します。

1. 連携ツール有無確認

連携ツール情報を各デバイス・ツール用 CIM Provider で管理しているプロバイダ管理情報から取得します。

連携ツール情報がある場合はアプリケーション名称と説明文テキストの内容を表示し、OK ボタンを活性にします。連携ツール情報がない場合は何も表示せず、ボタンは非活性にします。

サンプルソースの処理内容詳細

サンプルソースでは、RAID 用 CIM Provider の全デバイスの連携ツール情報を取得して表示します。

(1) RAID 用 CIM Provider 連携ツール表示

1. RAID 用 CIM Provider のプロバイダ管理情報取得メソッドを実行し、プロバイダ管理情報を取得する

2. メソッド実行結果が正常かつプロバイダ管理情報の接続デバイス情報が存在する場合、接続デバイス情報を取得する

メソッド実行結果が異常の場合、メソッド実行結果を表示して RAID 用 CIM Provider 連携ツール表示処理を終了する

3. WMI のクエリを実行して、PCI デバイス情報のデバイス名称(詳細)、詳細説明情報、製造元情報、デバイス ID を取得する

4. プロバイダ名をツリーの頂点ノードに表示する

5. 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合、PCI デバイス情報のデバイス名称(詳細)をツリー表示する

6. 接続デバイス情報の拡張情報をキーとバリューに分割して表示する

拡張情報がキーとバリューに分割できない場合は、拡張情報を表示しない

7. 接続デバイス情報からアプリケーション情報を取得して、アプリケーション名称と実行パス、説明文テキストパスを表示する

8. メソッド実行結果を表示する

連携ツール表示のサンプルソース

```
' 連携ツール表示: prtLinkToolInfo.vbs
Option Explicit

Dim objWMIService
Dim objOutParams
Dim objPnPSets, objPnP
Dim objDeviceInfo, objLinkAppInfo, objExtendedInfo
Dim strManualName, strDispManualName, strModulePath
Dim intExtendedPos
Dim strComputer

' スクリプトに引数がない場合はローカルホスト、ある場合は第1引数を接続先ホスト名とする
If WScript.Arguments.Count = 0 Then
    strComputer = "."
Else
    strComputer = WScript.Arguments.Item(0)
End If

' root/cimv2 に接続
Set objWMIService = GetObject("winmgmts:" & strComputer & "\root\CIMV2")

(1)
1.   ' プロバイダ管理情報取得メソッドを実行して戻り値を取得
Set objOutParams = objWMIService.ExecMethod("HITACHI_DevServiceRaid", "GetProviderManageInfo")
2.   ' メソッド実行結果が正常の場合
If objOutParams.ReturnValue = &H0 Then
    4.     ' プロバイダ名を表示
    Wscript.Echo objOutParams.DevProviderManageInfo.ProviderName
    ' プロバイダ管理情報の接続デバイス情報が存在する場合
    If IsNull(objOutParams.DevProviderManageInfo.DeviceInfo) = False Then
        ' 接続デバイス情報を取得
        For Each objDeviceInfo In objOutParams.DevProviderManageInfo.DeviceInfo
            ' 接続デバイス情報のアプリケーション情報が存在する場合
            If IsNull(objDeviceInfo.LinkAppInfo) = False Then
                ' OS から PCI デバイス情報を取得
                Set objPnPSet = objWMIService.ExecQuery _
                    ("SELECT Caption, DeviceID FROM Win32_PnPEntity WHERE DeviceID LIKE 'PCI%' AND ConfigManagerErrorCode = '0' AND Status = 'OK'")
                ' 接続デバイス情報と合致する PCI デバイス情報を検索
                For Each objPnP In objPnPSet
                    ' 接続デバイス情報のデバイス ID と PCI デバイス情報のデバイス ID が一致した場合
                    If objDeviceInfo.DeviceID = objPnP.DeviceID Then
                        ' デバイスリストにデバイス名称を表示
                        Wscript.Echo " " & objPnP.Caption
                        ' 拡張情報が存在する場合
                        If IsNull(objDeviceInfo.ExtendedInfo) = False Then
                            ' 拡張情報を取得
                            For Each objExtendedInfo In objDeviceInfo.ExtendedInfo
                                ' 拡張情報の切り出し位置を取得
                                intExtendedPos = InStr(objExtendedInfo, "?")
                                ' 拡張情報の切り出しが可能な場合
                                If intExtendedPos <> 0 Then
                                    ' 拡張情報を表示
                                    Wscript.Echo " " & Mid(objExtendedInfo, 1, intExtendedPos - 1) & ":" & Mid(objExtendedInfo, intExtendedPos + 1)
                                End If
                            Next
                        End If
                        ' アプリケーション情報を取得
                        For Each objLinkAppInfo In objDeviceInfo.LinkAppInfo
                            ' アプリケーション情報を表示
                            ' アプリケーション名称
                            Wscript.Echo "     ApplicationName: " & objLinkAppInfo.ApplicationName
                            ' 実行パス
                            Wscript.Echo "     ExecutePath: " & objLinkAppInfo.ExecutePath
                        End If
                    End If
                Next
            End If
        Next
    End If
    ' アプリケーション情報を取得
    For Each objLinkAppInfo In objDeviceInfo.LinkAppInfo
        ' アプリケーション情報を表示
        ' アプリケーション名称
        Wscript.Echo "     ApplicationName: " & objLinkAppInfo.ApplicationName
        ' 実行パス
        Wscript.Echo "     ExecutePath: " & objLinkAppInfo.ExecutePath
    End If
End If
```

```
    ' 説明文テキストパス
    Wscript.Echo "    DescriptionFilePath: " & objLinkAppInfo.DescriptionFilePath
    Next
    Exit For
End If
Next
End If
Next
End If
End If
' メソッド実行結果を表示
Wscript.echo "ReturnValue GetProviderManageInfo: " & Hex(objOutParams.ReturnValue)

Set objExtendedInfo = Nothing
Set objLinkAppInfo = Nothing
Set objDevInfo = Nothing
Set objPnP = Nothing
Set objPnPSet = Nothing
Set objOutParams = Nothing
Set objWMIService = Nothing
```


頭字語と略語

BIOS	Basic Input/Output System
BMC	Baseboard management controller
CIM	Common Information Model
CIMOM	Common Information Model Object Manager
CLI	Command Line Interface
EFI	extensible firmware interface
FC	Fibre Channel
F/W	Firmware
GB	gigabyte
GUI	Graphical User Interface
HRN	Hitachi RAID Navigator
Hz	Hertz
KB	Kilobyte
LAN	local area network
MAC	Media Access Control
MSM	MegaRAID Storage Manager
NMI	Non Maskable Interrupt
OS	operating system
RAID	Redundant Arrays of Inexpensive Disks
SAS	Serial Attached SCSI
SMI	System Management Interrupt
URL	Uniform Resource Locator
USB	Universal serial bus
VGA	video graphics array
VM	virtual machine
WMI	Windows Management Instrumentation
WWN	World Wide Name

◎ 株式会社 日立製作所
ITプラットフォーム事業本部

〒259-1392 神奈川県秦野市堀山下1番地

<http://www.hitachi.co.jp>