

---

## iStorage M シリーズ NAS オプション ソフトウェア

Virtual File Platform

## API リファレンス

## 対象製品

Virtual File Platform

5.5.1-00 以降

## 輸出時の注意

本製品（ソフトウェアを含む）は、外国為替及び外国貿易法で規定される規制貨物（または役務）に該当することがあります。

その場合、日本国外へ輸出する場合には日本国政府の輸出許可が必要です。

なお、輸出許可申請手続にあたり資料等が必要な場合には、お買い上げの販売店またはお近くの当社営業拠点にご相談ください。

## 商標類

Virtual File Platform は、株式会社日立製作所の登録商標です。

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

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

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

すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標がついた製品は、米国 Sun Microsystems, Inc. が開発したアーキテクチャに基づくものです。

UNIX は、The Open Group の米国ならびに他の国における登録商標です。

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

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

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

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

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

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

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

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

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

This product includes software developed by Andy Clark.

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



## 発行

2018 年 3 月（第 8 版）

# 目次

はじめに .....	13
対象読者 .....	14
マニュアルの構成 .....	14
マニュアル体系 .....	14
このマニュアルでの表記 .....	15
このマニュアルで使用する記号 .....	15
このマニュアルで使用する構文要素 .....	16
KB（キロバイト）などの単位表記について .....	16
 1. API の概要 .....	17
1.1 API を使用してできること .....	18
1.2 システム構成 .....	18
1.3 使用上の注意事項 .....	18
1.4 リソースとプロパティ .....	18
1.5 HTTP メソッド .....	19
1.6 リクエストの実行方法 .....	20
1.6.1 GET メソッドを使用する場合 .....	20
1.6.2 PUT メソッドを使用する場合 .....	21
1.6.3 POST メソッドを使用する場合 .....	21
1.6.4 DELETE メソッドを使用する場合 .....	22
1.6.5 HEAD メソッドを使用する場合 .....	22
1.7 HTTP ステータスコード .....	23
1.8 入出力形式 .....	24
1.9 クエリパラメーター .....	24
1.10 リクエストヘッダー .....	25
1.11 レスポンスヘッダー .....	26
1.12 エラー発生時のレスポンス XML .....	27
 2. アクセスと認証 .....	29
2.1 HVFP にアクセスするための URI とアカウント .....	30
2.2 URI についての考慮事項 .....	30
2.2.1 URI の文字長 .....	30
2.2.2 特定の文字または値のエンコーディング .....	30
2.2.3 コマンドラインでの引用符の使用 .....	31
2.2.4 マルチバイト文字の使用 .....	31
2.3 認証 .....	31

2.4 公開鍵証明書 .....	31
------------------	----

### 3. リソース..... 33

3.1 リソースの一覧 .....	34
3.2 APIAccount .....	34
3.2.1 APIAccount リソースの概要 .....	34
3.2.2 APIAccount リソース .....	34
3.2.2.1 GET メソッドで API の管理者アカウントの情報を取得する .....	34
3.2.2.2 POST メソッドで API の管理者アカウントのパスワードを変更する .....	35
3.3 CIFSShare .....	37
3.3.1 CIFSShare リソースの概要 .....	37
3.3.2 CIFSShare リソース .....	37
3.3.2.1 GET メソッドですべての CIFS 共有の情報を取得する .....	37
3.3.2.2 PUT メソッドで CIFS 共有を追加する .....	44
3.3.3 CIFSShares/ <Base64 でエンコードされた CIFS 共有名> リソース .....	66
3.3.3.1 GET メソッドで CIFS 共有の情報を取得する .....	66
3.3.3.2 POST メソッドで CIFS 共有の情報を変更する .....	68
3.3.3.3 DELETE メソッドで指定した CIFS 共有を削除する .....	78
3.3.3.4 HEAD メソッドで指定した CIFS 共有が存在するかどうかを確認する .....	78
3.4 FCPath .....	79
3.4.1 FCPath リソースの概要 .....	79
3.4.2 FCPath リソース .....	79
3.4.2.1 GET メソッドで指定した FC パスの情報を取得する .....	79
3.5 FileSystem .....	82
3.5.1 FileSystem リソースの概要 .....	82
3.5.2 FileSystem リソース .....	82
3.5.2.1 GET メソッドでファイルシステムの情報を取得する .....	83
3.5.2.2 PUT メソッドでファイルシステムを作成する .....	89
3.5.3 FileSystems/ <ファイルシステム名> リソース .....	96
3.5.3.1 GET メソッドで指定したファイルシステムの情報を取得する .....	96
3.5.3.2 POST メソッドで指定したファイルシステムの情報を編集する .....	99
3.5.3.3 DELETE メソッドで指定したファイルシステムを削除する .....	104
3.5.3.4 HEAD メソッドで指定したファイルシステムが存在するかどうかを確認する .....	104
3.5.4 FileSystems/ <ファイルシステム名> /MountSetting リソース .....	104
3.5.4.1 GET メソッドで指定したファイルシステムのマウント設定の情報を取得する .....	104
3.5.4.2 POST メソッドで指定したファイルシステムのマウント設定の情報を編集する .....	107
3.6 Hardware .....	109
3.6.1 Hardware リソースの概要 .....	109
3.6.2 Hardware リソース .....	109
3.6.2.1 GET メソッドでハードウェア情報を取得する .....	109
3.7 LU .....	118
3.7.1 LU リソースの概要 .....	118
3.7.2 LU リソース .....	118
3.7.2.1 GET メソッドで LU の情報を取得する .....	118
3.8 NetworkInterface .....	122
3.8.1 NetworkInterface リソースの概要 .....	122
3.8.2 NetworkInterface リソース .....	123
3.8.2.1 GET メソッドでネットワークインターフェースの情報を取得する .....	123
3.8.2.2 PUT メソッドでネットワークインターフェースを追加する .....	125
3.8.3 NetworkInterfaces/ <ネットワークインターフェース名> リソース .....	128
3.8.3.1 GET メソッドでネットワークインターフェースの情報を取得する .....	128
3.8.3.2 POST メソッドでネットワークインターフェースの情報を変更する .....	130
3.8.3.3 HEAD メソッドで指定したネットワークインターフェースが存在するかどうかを確認する .....	

3.9 NFSShare	133
3.9.1 NFSShare リソースの概要	133
3.9.2 NFSShare リソース	134
3.9.2.1 GET メソッドですべての NFS 共有の情報を取得する	134
3.9.2.2 PUT メソッドで NFS 共有を追加する	137
3.9.3 NFSShares/ < NFS 共有ディレクトリ名 > リソース	147
3.9.3.1 GET メソッドで指定した NFS 共有の情報を取得する	148
3.9.3.2 POST メソッドで NFS 共有の情報を編集する	149
3.9.3.3 DELETE メソッドで指定した NFS 共有を削除する	154
3.9.3.4 HEAD メソッドで指定した NFS 共有が存在するかどうか確認する	154
3.10 ProcessingNode	154
3.10.1 ProcessingNode リソースの概要	154
3.10.2 ProcessingNode リソース	155
3.10.2.1 GET メソッドでクラスタおよびノードの情報を取得する	155
3.11 Routing	159
3.11.1 Routing リソースの概要	159
3.11.2 Routing リソース	159
3.11.2.1 GET メソッドで指定したルーティング情報を取得する	159
3.11.2.2 POST メソッドでルーティング情報を追加または削除する	162
3.12 SystemController	166
3.12.1 SystemController リソースの概要	166
3.12.2 SystemController リソース	167
3.12.2.1 GET メソッドでサービスまたは OS の再起動が必要かどうかを確認する	167
3.12.2.2 POST メソッドでサービスまたは OS を再起動する	168
3.12.3 SystemController/OS リソース	170
3.12.3.1 GET メソッドで OS の起動状態を確認する	170
3.12.3.2 POST メソッドでノード上の OS を再起動または停止する	171
3.13 VirtualServer	172
3.13.1 VirtualServer リソースの概要	172
3.13.2 VirtualServer リソース	172
3.13.2.1 GET メソッドで Virtual Server の情報を取得する	172
<b>付録 A API の使用例</b>	<b>175</b>
A.1 クラスタおよびノードの情報を取得する例	176
A.2 ハードウェア状態を監視する例	177
A.3 ネットワークを設定する例	180
A.4 ファイルシステムを作成する例	183
A.5 ファイルシステムを削除する例	195
A.6 OS を停止する例	197





# 図目次

図 1-1：システム構成 .....	18
--------------------	----





# 表目次

表 1-1: サポートする HTTP メソッド	19
表 1-2: 正常終了した場合の HTTP ステータスコード	23
表 1-3: 異常終了した場合の HTTP ステータスコード	23
表 1-4: サポートする MIME タイプ	24
表 1-5: サポートするクエリパラメーター	25
表 1-6: リクエストヘッダーの内容	25
表 1-7: レスポンスヘッダーの内容	26
表 1-8: エラー発生時のレスポンス XML の構造	27
表 2-1: パーセントエンコードする文字	30
表 3-1: リソースの使用目的	34
表 3-2: APIAccount リソースに使用できる HTTP メソッド	34
表 3-3: APIAccount リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	35
表 3-4: APIAccount リソースに GET メソッドをリクエストしたときのプロパティ	35
表 3-5: APIAccount リソースに GET メソッドをリクエストしたときのステータスコード	35
表 3-6: APIAccount リソースに POST メソッドをリクエストするときの XML の構造	36
表 3-7: APIAccount リソースに POST メソッドをリクエストするときのプロパティ	36
表 3-8: APIAccount リソースに POST メソッドをリクエストしたときのステータスコード	36
表 3-9: CIFSShare リソースに使用できる HTTP メソッド	37
表 3-10: CIFSShare リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	37
表 3-11: CIFSShare リソースに GET メソッドをリクエストしたときのプロパティ	39
表 3-12: CIFSShare リソースに GET メソッドをリクエストしたときのステータスコード	44
表 3-13: CIFSShare リソースに PUT メソッドをリクエストするときの XML の構造	44
表 3-14: CIFSShare リソースに PUT メソッドをリクエストするときのプロパティ	47
表 3-15: CIFSShare リソースに PUT メソッドをリクエストしたときのステータスコード	66
表 3-16: CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	66
表 3-17: CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに GET メソッドをリクエストしたときのステータスコード	68
表 3-18: CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに POST メソッドをリクエストするときの XML の構造	68
表 3-19: CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに POST メソッドをリクエストするときのプロパティ	69
表 3-20: CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに POST メソッドをリクエストしたときのステータスコード	78

表 3-21: CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに DELETE メソッドをリクエストしたときのステータスコード	78
表 3-22: CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに HEAD メソッドをリクエストしたときのステータスコード	79
表 3-23: FCPath リソースに使用できる HTTP メソッド	79
表 3-24: FCPath リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	79
表 3-25: FCPath リソースに GET メソッドをリクエストしたときのプロパティ	80
表 3-26: FCPath リソースに GET メソッドをリクエストしたときのステータスコード	82
表 3-27: FileSystem リソースに使用できる HTTP メソッド	82
表 3-28: FileSystem リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	83
表 3-29: FileSystem リソースに GET メソッドをリクエストしたときのプロパティ	84
表 3-30: FileSystem リソースに GET メソッドをリクエストしたときのステータスコード	89
表 3-31: FileSystem リソースに PUT メソッドをリクエストするときの XML の構造	90
表 3-32: FileSystem リソースに PUT メソッドをリクエストするときのプロパティ	91
表 3-33: FileSystem リソースに PUT メソッドをリクエストしたときのステータスコード	96
表 3-34: FileSystems/ < ファイルシステム名 > リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	96
表 3-35: FileSystems/ < ファイルシステム名 > リソースに GET メソッドをリクエストしたときのステータスコード	99
表 3-36: FileSystems/ < ファイルシステム名 > リソースに POST メソッドをリクエストするときの XML の構造	100
表 3-37: FileSystems/ < ファイルシステム名 > リソースに POST メソッドをリクエストするときのプロパティ	100
表 3-38: FileSystems/ < ファイルシステム名 > リソースに POST メソッドをリクエストしたときのステータスコード	104
表 3-39: FileSystems/ < ファイルシステム名 > リソースに DELETE メソッドをリクエストしたときのステータスコード	104
表 3-40: FileSystems/ < ファイルシステム名 > リソースに HEAD メソッドをリクエストしたときのステータスコード	104
表 3-41: FileSystems/ < ファイルシステム名 > /MountSetting リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	104
表 3-42: FileSystems/ < ファイルシステム名 > /MountSetting リソースに GET メソッドをリクエストしたときのプロパティ	105
表 3-43: FileSystems/ < ファイルシステム名 > /MountSetting リソースに GET メソッドをリクエストしたときのステータスコード	107
表 3-44: FileSystems/ < ファイルシステム名 > /MountSetting リソースに POST メソッドをリクエストするときの XML の構造	107
表 3-45: FileSystems/ < ファイルシステム名 > /MountSetting リソースに POST メソッドをリクエストするときのプロパティ	107
表 3-46: FileSystems/ < ファイルシステム名 > /MountSetting リソースに POST メソッドをリクエストしたときのステータスコード	108
表 3-47: Hardware リソースに使用できる HTTP メソッド	109
表 3-48: Hardware リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	109
表 3-49: Hardware リソースに GET メソッドをリクエストしたときのプロパティ	111
表 3-50: Hardware リソースに GET メソッドをリクエストしたときのステータスコード	117
表 3-51: LU リソースに使用できる HTTP メソッド	118
表 3-52: LU リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	118
表 3-53: LU リソースに GET メソッドをリクエストしたときのプロパティ	119
表 3-54: LU リソースに GET メソッドをリクエストしたときのステータスコード	122
表 3-55: NetworkInterface リソースに使用できる HTTP メソッド	122

表 3-56: NetworkInterface リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	123
表 3-57: NetworkInterface リソースに GET メソッドをリクエストしたときのプロパティ	124
表 3-58: NetworkInterface リソースに GET メソッドをリクエストしたときのステータスコード	125
表 3-59: NetworkInterface リソースに PUT メソッドをリクエストするときの XML の構造	126
表 3-60: NetworkInterface リソースに PUT メソッドをリクエストするときのプロパティ	126
表 3-61: NetworkInterface リソースに PUT メソッドをリクエストしたときのステータスコード	128
表 3-62: NetworkInterfaces/ <ネットワークインターフェース名>リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	129
表 3-63: NetworkInterfaces/ <ネットワークインターフェース名>リソースに GET メソッドをリクエストしたときのステータスコード	130
表 3-64: NetworkInterfaces/ <ネットワークインターフェース名>リソースに POST メソッドをリクエストするときの XML の構造	130
表 3-65: NetworkInterfaces/ <ネットワークインターフェース名>リソースに POST メソッドをリクエストするときのプロパティ	130
表 3-66: NetworkInterfaces/ <ネットワークインターフェース名>リソースに POST メソッドをリクエストしたときのステータスコード	133
表 3-67: NetworkInterfaces/ <ネットワークインターフェース名>リソースに HEAD メソッドをリクエストしたときのステータスコード	133
表 3-68: NFSShare リソースに使用できる HTTP メソッド	133
表 3-69: NFSShare リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	134
表 3-70: NFSShare リソースに GET メソッドをリクエストしたときのプロパティ	135
表 3-71: NFSShare リソースに GET メソッドをリクエストしたときのステータスコード	137
表 3-72: NFSShare リソースに PUT メソッドをリクエストするときの XML の構造	137
表 3-73: NFSShare リソースに PUT メソッドをリクエストするときのプロパティ	139
表 3-74: NFSShare リソースに PUT メソッドをリクエストしたときのステータスコード	147
表 3-75: NFSShares/ < NFS 共有ディレクトリ名>リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	148
表 3-76: NFSShares/ < NFS 共有ディレクトリ名>リソースに GET メソッドをリクエストしたときのステータスコード	149
表 3-77: NFSShares/ < NFS 共有ディレクトリ名>リソースに POST メソッドをリクエストするときの XML の構造	150
表 3-78: NFSShares/ < NFS 共有ディレクトリ名>リソースに POST メソッドをリクエストするときのプロパティ	150
表 3-79: NFSShares/ < NFS 共有ディレクトリ名>リソースに POST メソッドをリクエストしたときのステータスコード	154
表 3-80: NFSShares/ < NFS 共有ディレクトリ名>リソースに DELETE メソッドをリクエストしたときのステータスコード	154
表 3-81: NFSShares/ < NFS 共有ディレクトリ名>リソースに HEAD メソッドをリクエストしたときのステータスコード	154
表 3-82: ProcessingNode リソースに使用できる HTTP メソッド	155
表 3-83: ProcessingNode リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	155
表 3-84: ProcessingNode リソースに GET メソッドをリクエストしたときのプロパティ	156
表 3-85: ProcessingNode リソースに GET メソッドをリクエストしたときのステータスコード	159
表 3-86: Routing リソースに使用できる HTTP メソッド	159
表 3-87: Routing リソースに GET メソッドをリクエストしたときのレスポンス XML の構造	160
表 3-88: Routing リソースに GET メソッドをリクエストしたときのプロパティ	160
表 3-89: Routing リソースに GET メソッドをリクエストしたときのステータスコード	162
表 3-90: Routing リソースに POST メソッドをリクエストするときの XML の構造	162
表 3-91: Routing リソースに POST メソッドをリクエストするときのプロパティ	163

表 3-92: Routing リソースに POST メソッドをリクエストしたときのステータスコード .....	166
表 3-93: SystemController リソースに使用できる HTTP メソッド .....	166
表 3-94: SystemController リソースに GET メソッドをリクエストしたときのレスポンス XML の構造 .....	167
表 3-95: SystemController リソースに GET メソッドをリクエストしたときのプロパティ .....	167
表 3-96: SystemController リソースに GET メソッドをリクエストしたときのステータスコード .....	168
表 3-97: SystemController リソースに POST メソッドをリクエストするときの XML の構造 .....	168
表 3-98: SystemController リソースに POST メソッドをリクエストするときのプロパティ .....	169
表 3-99: SystemController リソースに POST メソッドをリクエストしたときのステータスコード .....	170
表 3-100: SystemController/OS リソースに GET メソッドをリクエストしたときのレスポンス XML の構造 ...	170
表 3-101: SystemController/OS リソースに GET メソッドをリクエストしたときのプロパティ .....	170
表 3-102: SystemController/OS リソースに GET メソッドをリクエストしたときのステータスコード .....	171
表 3-103: SystemController/OS リソースに POST メソッドをリクエストするときの XML の構造 .....	171
表 3-104: SystemController/OS リソースに POST メソッドをリクエストするときのプロパティ .....	171
表 3-105: SystemController/OS リソースに POST メソッドをリクエストしたときのステータスコード .....	171
表 3-106: VirtualServer リソースに使用できる HTTP メソッド .....	172
表 3-107: VirtualServer リソースに GET メソッドをリクエストしたときのレスポンス XML の構造 .....	172
表 3-108: VirtualServer リソースに GET メソッドをリクエストしたときのプロパティ .....	172
表 3-109: VirtualServer リソースに GET メソッドをリクエストしたときのステータスコード .....	173
表 A-1: クラスタおよびノードの情報を取得する例 .....	176
表 A-2: ハードウェア状態を監視する例 .....	177
表 A-3: ネットワークを設定する例 .....	180
表 A-4: ファイルシステムを作成する例 .....	184
表 A-5: ファイルシステムを削除する例 .....	195
表 A-6: OS を停止する例 .....	197



# はじめに

このマニュアルは、Virtual File Platform（HVFP）の API の使用方法について説明したものです。

- [対象読者](#)
- [マニュアルの構成](#)
- [マニュアル体系](#)
- [このマニュアルでの表記](#)
- [このマニュアルで使用する記号](#)
- [このマニュアルで使用する構文要素](#)
- [KB（キロバイト）などの単位表記について](#)

# 対象読者

このマニュアルは、API を使用して、HVFP の管理アプリケーションを開発する方にお読みいただくことを前提に説明しています。

また、次の知識をお持ちであることを前提にしています。

- HVFP の運用および管理に関する知識
- REST インターフェースを使用したアプリケーションプログラミングに関する知識

# マニュアルの構成

このマニュアルは、次に示す章と付録から構成されています。

章	内容
1. API の概要	HVFP の API を使用する場合はシステム構成、リソースとプロパティ、HTTP メソッド、リクエストの実行方法、HTTP ステータスコード、リクエストおよびレスポンス時の入出力形式、クエリパラメーター、リクエストヘッダーおよびレスポンスヘッダーなどについて説明しています。
2. アクセスと認証	HVFP にアクセスするための URL、URI についての考慮事項、認証情報の指定方法、および SSL 証明書について説明しています。
3. リソース	それぞれのリソースで使用できる HTTP メソッドの種類、各メソッドのリクエスト XML とレスポンス XML の構造、プロパティの詳細、リクエスト先の URI、およびメソッドが終了したときに返されるステータスコードについて説明しています。
A. API の使用例	API の使用方法を例を挙げて説明しています。

# マニュアル体系

HVFP のマニュアル体系を次に示します。

マニュアル名	内容
Virtual File Platform ファーストステップガイド (IF201A)	HVFP をセットアップする前に検討しておくべきこと、および、セットアップの手順について説明しています。
Virtual File Platform システム構成ガイド (IF202A)	HVFP を運用するために、最初にお読みいただくマニュアルです。HVFP の運用を開始する前に理解または検討しておいていただきたいことや、外部サーバの環境設定などについて説明しています。
Virtual File Platform セットアップガイド (IF203A)	HVFP のセットアップ方法について説明しています。仮想サーバで HVFP を運用する場合は、「仮想サーバ環境セットアップガイド」(IF204A) をお読みください。
Virtual File Platform 仮想サーバ環境セットアップガイド (IF204A)	HVFP での Virtual Server のセットアップ方法について説明しています。
Virtual File Platform 仮想サーバ環境セットアップガイド別紙 (IF218A)	HVFP における Virtual Server の性能に関する説明をしています。
Virtual File Platform ユーザーズガイド (IF205A)	HVFP を運用するために必要な手順や GUI リファレンスなどを説明しています。

マニュアル名	内容
Virtual File Platform ファイルアクセス (CIFS/NFS) ユーザーズガイド (IF206A)	CIFS または NFS クライアントから、HVFP の CIFS サービスまたは NFS サービスを利用するに当たって、事前に知っておいていただきたいことや、注意する必要があることについて説明しています。
Virtual File Platform ファイルアクセス (Quota) ユーザーズガイド (IF207A)	ファイルシステムやディレクトリに <b>Quota</b> を設定する際に、事前に知っておいていただきたいことや、注意する必要があることについて説明しています。
Virtual File Platform トラブルシューティングガイド (IF208A)	HVFP の障害対策を説明しています。
Virtual File Platform トラブルシューティングガイド別紙 (IF209A)	HVFP のソフトウェア障害の回復手順について説明しています。
Virtual File Platform インストールガイド (IF210A)	HVFP のインストール方法について説明しています。
Virtual File Platform コマンドリファレンス (IF211A)	HVFP で使用できるコマンドの文法について説明しています。
Virtual File Platform API リファレンス (IF212A) (このマニュアル)	HVFP の API の使用方法について説明しています。
Virtual File Platform メッセージリファレンス (IF213A)	HVFP のメッセージについて説明しています。
Virtual File Platform メッセージリファレンス別紙 (IF214A)	HVFP の SNMP Trap にて通知されるメッセージについて説明しています。
Virtual File Platform ESMPRO 通報設定 (IF215A)	ESMPRO と連携して通報を行うための設定方法について説明しています。
Virtual File Platform BackupRestore 機能補足資料 (NetBackup) (IF216A)	NetBackup のマニュアルの理解を補助するためのものです。
Virtual File Platform システム動作情報のグラフ化手順書 (IF217A)	HVFP のシステム動作情報をグラフ化する手順について説明しています。

## このマニュアルでの表記

このマニュアルでは、製品の名称を省略して表記しています。このマニュアルでの表記と、製品の正式名称または意味を次の表に示します。

このマニュアルでの表記	製品名称または意味
HVFP	Virtual File Platform
Windows	Microsoft(R) Windows(R) Operating System

## このマニュアルで使用する記号

このマニュアルでは、次に示す記号を使用しています。

記号	意味
[ ]	画面，メニュー，ボタン，キーボードのキーなどを示します。 (例) [CIFS Service Management] ページ
< >	可変値であることを示します。 (例) <ホスト名>.<ポート番号> 実際のホスト名が「host0」，ポート番号が「1024」の場合，「host0.1024」と指定することを示します。
...	この記号の直前に示された項目を繰り返して複数指定できます。項目を複数指定する場合は，項目の区切りにコンマ (,) を使用します。 (例) A,B,... 「A の後ろに，B を複数指定できる」ことを示します。

## このマニュアルで使用する構文要素

このマニュアルで使用する構文要素（設定値やファイル名などに指定できる値）の種類を，次のように定義します。

種類	定義
数字	0 ～ 9
英数字	A ～ Z a ～ z 0 ～ 9

注 すべての半角で指定してください。

## KB（キロバイト）などの単位表記について

1KB（キロバイト），1MB（メガバイト），1GB（ギガバイト），1TB（テラバイト），1PB（ペタバイト）はそれぞれ 1,024 バイト， $1,024^2$  バイト， $1,024^3$  バイト， $1,024^4$  バイト， $1,024^5$  バイトです。



# API の概要

Virtual File Platform (HVFP) の API は、HVFP を管理するアプリケーションプログラムを作成するためのインターフェースです。

この章では、HVFP の API を使用する場合のシステム構成、リソースとプロパティ、HTTP メソッド、リクエストの実行方法、HTTP ステータスコード、リクエストおよびレスポンス時の入出力形式、クエリパラメーター、リクエストヘッダーおよびレスポンスヘッダーなどについて説明します。

- [1.1 API を使用してできること](#)
- [1.2 システム構成](#)
- [1.3 使用上の注意事項](#)
- [1.4 リソースとプロパティ](#)
- [1.5 HTTP メソッド](#)
- [1.6 リクエストの実行方法](#)
- [1.7 HTTP ステータスコード](#)
- [1.8 入出力形式](#)
- [1.9 クエリパラメーター](#)
- [1.10 リクエストヘッダー](#)
- [1.11 レスポンスヘッダー](#)
- [1.12 エラー発生時のレスポンス XML](#)

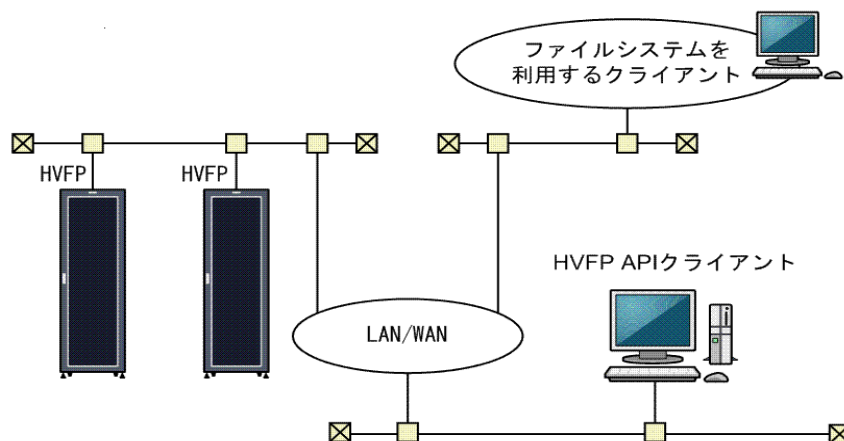
## 1.1 API を使用してできること

HVFP では、HTTP REST (REpresentational State Transfer) インターフェース形式の API を提供しています。API を使用することで、HVFP をネットワークに接続するための設定や、ファイルシステムの作成などの操作が可能です。

## 1.2 システム構成

HVFP の API を使用する場合のシステム構成を次の図に示します。

図 1-1: システム構成



### HVFP

HVFP は、ネットワーク上のクライアントに対してファイルシステムのサービスを提供するためのシステムです。

### HVFP API クライアント

HVFP API クライアントは、HVFP に API を使用して接続し、HVFP のノードを管理します。HVFP API クライアントは、API を使用してリクエストを送信し、処理結果（レスポンス）を API を通じて受信します。HVFP API クライアントは、HVFP のノードと通信できるように設定しておく必要があります。

### ファイルシステムを利用するクライアント

CIFS 共有または NFS 共有を使用してファイルシステムを利用できます。

## 1.3 使用上の注意事項

HVFP の API は、リクエストの処理が 30 分以内に完了しない場合、タイムアウトによって処理が中断します。

## 1.4 リソースとプロパティ

HVFP の API を使用すると、リソースを管理できます。リソースには、FileSystem, ProcessingNode, Routing などがあります。

リソースはプロパティを持っていて、プロパティの値でリソースを定義します。

それ自身がリソースとして扱われるプロパティもあります。

リソースを識別するには、URI を使用します。例えば、次の URI は `system.example.com` という名称の HVFP システムにある、`fs01` という名称の `FileSystem` リソースを表しています。

`https://system.example.com:9090/mapi/FileSystems/fs01`

プロパティにはデータ型があります。プロパティのデータ型には、`String`、`Integer` および `Boolean` があります。例えば、`FileSystem` リソースの `fileName` プロパティのデータ型は `String` です。

## 1.5 HTTP メソッド

HVFP の API がサポートする HTTP/1.1 のメソッドを次の表に示します。

表 1-1: サポートする HTTP メソッド

メソッド	説明
GET	リクエスト URI が示すリソースの情報を取得するためのメソッドです。
PUT	リクエスト URI が示すリソースに対して情報を追加するためのメソッドです。
POST	リクエスト URI が示すリソースの情報を変更するためのメソッドです。
DELETE	リクエスト URI が示すリソースを削除するためのメソッドです。
HEAD	リクエスト URI が示すリソースのメタ情報を取得するためのメソッドです。 レスポンスにメッセージボディが返されない（メタ情報だけが含まれる）点だけが GET メソッドと異なります。
OPTIONS	リクエスト URI が示すリソースで利用できるメソッドを確認するためのメソッドです。

HVFP の API では、すべてのリソース URI に対して `OPTIONS` メソッドを使用できます。特定のリソース URI に対して `OPTIONS` メソッドをリクエストすると、利用できる HTTP メソッドの種類と、それぞれのメソッドでサポートされている MIME タイプが返されます。

`OPTIONS` メソッドをリクエストすると、レスポンスには次の内容が含まれます。

- Allow ヘッダー

リクエスト URI でサポートされている HTTP メソッドの一覧です。

例えば、`GET`、`HEAD` および `POST` メソッドをサポートしている URI に対して `OPTIONS` メソッドをリクエストすると、レスポンスには次のような `Allow` ヘッダーが含まれます。

`Allow: OPTIONS, GET, HEAD, POST`

- WADL

リクエスト URI について各 HTTP メソッドでサポートされている MIME タイプが WADL で表されます。WADL は、Web サービスの仕様を定義する XML ドキュメントです。

WADL は、`OPTIONS` メソッドのリクエストに対するレスポンスボディに設定されます。

`FileSystem` リソースに対して `OPTIONS` メソッドをリクエストした場合の WADL XML 出力の例を次に示します。

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<application xmlns="http://research.sun.com/wadl/2006/10">
  <doc jersey:generatedBy="Jersey: 1.1.5 01/21/2011 03:50 PM"
  xmlns:jersey="http://jersey.dev.java.net/" />
  <resources base="https://nec.example.com:9090/mapi/">
    <resource path="FileSystems">
      <method id="getFileSystems" name="GET">
        <response>
          <representation mediaType="application/xml"/>
        </response>
      </method>
    </resource>
  </resources>
</application>
```

```

    </response>
  </method>
  <method id="createFileSystem" name="PUT">
    <request>
      <representation mediaType="application/xml"/>
    </request>
    <response>
      <representation mediaType="*/*/">
    </response>
  </method>
</resource>
</resources>
</application>

```

## 1.6 リクエストの実行方法

ここでは、REST インターフェースを使用して各メソッドをリクエストする方法について説明します。このマニュアルでは、curl コマンド（バージョン 7.18.2）を使用しています。

curl コマンドで指定するユーザー名とパスワードについては、「[2.3 認証](#)」を参照してください。

### 1.6.1 GET メソッドを使用する場合

GET メソッドは、リソースの情報を取得するために使用します。リソースの情報は、XML 形式で返されます。

fs という名称のファイルシステムの情報を取得する場合、次のように curl コマンドを実行します。

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIP
アドレス>:9090/mapi/FileSystems/fs?prettyprint" -k -X GET -H "Accept:
application/xml"
```

curl コマンドを実行すると、次のようにステータスコード、ヘッダー、およびリソースの情報が表示されます。

```

HTTP/1.1 200 OK
Date: Fri, 10 Oct 2014 05:26:51 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 734
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get file system details
Connection: close
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FileSystems>
  <FileSystem>
    <fileName>fs</fileName>
    <isLvmEnabled>true</isLvmEnabled>
    <isLvmStripingEnabled>false</isLvmStripingEnabled>
    <stripeSize>64</stripeSize>
    <aclType>Advanced ACL</aclType>
    <isBypassEnabled>true</isBypassEnabled>
    <WormSetting>
      <maxRetention>10950-0-0</maxRetention>
      <minRetention>0-0-0</minRetention>
      <isAutoCommitEnabled>false</isAutoCommitEnabled>
      <commitModeSetting></commitModeSetting>
      <autoCommitPeriod></autoCommitPeriod>
      <defaultRetention></defaultRetention>
    </WormSetting>
    <LUs>
      <LU>
        <deviceFileName>lu0000</deviceFileName>
      </LU>
      <LU>
        <deviceFileName>lu0001</deviceFileName>
      </LU>
    </LUs>
  </FileSystem>
</FileSystems>

```

```

        </LUs>
    </FileSystem>
</FileSystems>

```

## 1.6.2 PUT メソッドを使用する場合

PUT メソッドは、リソースに情報を追加するために使用します。リクエストする内容を XML 形式のファイルで作成してから、curl コマンドを実行します。

fs という名称のファイルシステムで、WORM を設定し、LU を追加する場合、次のように指定します。

1. 次の内容を、fscreate.xml というファイル名で作成します。

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FileSystem>
    <fileName>fs</fileName>
    <aclType>Advanced ACL</aclType>
    <isLvmEnabled>true</isLvmEnabled>
    <stripeSize>64</stripeSize>
    <isLvmStripingEnabled>true</isLvmStripingEnabled>
    <WormSetting>
        <maxRetention>10950-0-0</maxRetention>
        <minRetention>0-0-0</minRetention>
        <isAutoCommitEnabled>false</isAutoCommitEnabled>
    </WormSetting>
    <LUs>
        <LU>
            <deviceFileName>lu0004</deviceFileName>
        </LU>
        <LU>
            <deviceFileName>lu0005</deviceFileName>
        </LU>
    </LUs>
</FileSystem>

```

2. curl コマンドを実行します。

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems" -k -d @fscreate.xml -X PUT -H
"Content-Type: application/xml"

```

curl コマンドを実行すると、ステータスコードおよびヘッダーが次のように表示されます。

```

HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 07:56:42 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Create file system
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1

```

## 1.6.3 POST メソッドを使用する場合

POST メソッドは、リソースの情報を変更するために使用します。リクエストする内容を XML 形式のファイルで作成してから、curl コマンドを実行します。

fs という名称のファイルシステムの容量を拡張して WORM の属性を変更する場合は、次のように

指定します。

1. 次の内容を fsmod.xml というファイル名で作成します。

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FileSystem>
    <WormSetting>
        <isAutoCommitEnabled>true</isAutoCommitEnabled>
        <commitModeSetting>manual</commitModeSetting>
        <autoCommitPeriod>0-0-15</autoCommitPeriod>
        <defaultRetention>10950-0-0</defaultRetention>
    </WormSetting>

```

```

<addLUs>
  <LU>
    <deviceFileName>lu0006</deviceFileName>
  </LU>
  <LU>
    <deviceFileName>lu0007</deviceFileName>
  </LU>
</addLUs>
</FileSystem>

```

2. curl コマンドを実行します。

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems/fs/" -k -d @fsmod.xml -X POST -H
"Content-Type: application/xml"
curl コマンドを実行すると、ステータスコードおよびヘッダーが次のように表示されます。
HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 07:57:20 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 2
X-NODE-OperationList: [1]Expand file system, [2]Edit file system
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1

```

## 1.6.4 DELETE メソッドを使用する場合

DELETE メソッドは、リソースの情報を削除するために使用します。

fs という名称のファイルシステムを削除する場合、次のように curl コマンドを実行します。

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems/fs/" -k -X DELETE -H "Content-Type: application/xml"

```

curl コマンドを実行すると、ステータスコードおよびヘッダーが次のように表示されます。

```

HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 05:39:27 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Delete file system
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1

```

## 1.6.5 HEAD メソッドを使用する場合

HEAD メソッドは、リソースが存在するかどうかを確認するために使用します。

fs という名称のファイルシステムが存在するかどうかを確認する場合、次のように curl コマンドを実行します。

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems/fs/" -k -X HEAD -H "Content-Type: application/xml"

```

curl コマンドを実行し、リソースが存在した場合は、ステータスコードおよびヘッダーが次のように表示されます。

```

HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 05:39:11 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 685
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get file system details
Connection: close
Content-Type: application/xml

```

## 1.7 HTTP ステータスコード

HVFP API クライアントからのリクエストに対して、HVFP は HTTP ステータスコードを返します。

メソッドの実行に成功した場合、HVFP はステータスコードを生成し、API を通じてレスポンスとして返します。なお、GET メソッドの場合は、ステータスコードに加えて、レスポンス XML を返します。GET メソッドのリクエストに対して返されたレスポンス XML は、URI が示すリソースについての詳細情報を示します。

リクエストが正常終了した場合のステータスコードを次の表に示します。

表 1-2：正常終了した場合の HTTP ステータスコード

ステータスコード		メソッド	説明
200	OK	すべて	HVFP は、リソースの作成、情報取得、存在確認、変更および削除に成功しました。
202	Accepted	すべて	HVFP の処理を受け付けましたが、このステータスコードが返された時点では処理の完了は保証されません。 しばらく待ってから、リソースの状態などを確認して、処理が正常に完了していることを確認してください。

メソッドの実行に失敗した場合、HVFP は、API を通じてレスポンスヘッダーに設定されるエラー情報と一緒に、エラーの種類を示すステータスコードを返します。エラーの場合に返されるステータスコードは、失敗の原因に基づいて決定されます。

リクエストが異常終了した場合のステータスコードを次の表に示します。この表で説明しているステータスコードのほかにも、RFC2068 で定義されているステータスコードが返される場合があります。

表 1-3：異常終了した場合の HTTP ステータスコード

ステータスコード		メソッド	説明
302	FOUND	すべて	リクエストが有効ではありませんでした。主に次の理由が考えられます。 <ul style="list-style-type: none"><li>• api-auth Cookie がありません。</li><li>• 無効な証明書が指定されています。</li></ul>
400	BAD_REQUEST	すべて	リクエストが有効ではありませんでした。主に次の理由が考えられます。 <ul style="list-style-type: none"><li>• リクエスト中に必須のクエリパラメーターがありません。</li><li>• PUT または POST メソッドをリクエストした場合<ul style="list-style-type: none"><li>・ リクエストボディに必須のプロパティがありません。</li><li>・ リクエストボディに無効なプロパティが含まれています。</li><li>・ リクエストボディに不正な形式の XML が含まれています。</li><li>・ データ型チェックに失敗しました。</li><li>・ リクエストボディに不正な値が含まれています。</li></ul></li><li>• リクエスト中の Content-Type または Accept ヘッダーに、存在しない MIME タイプが指定されています。</li></ul>
401	UNAUTHORIZED	すべて	リクエストヘッダーに指定したユーザーアカウントが認証で拒否されました。
403	FORBIDDEN	GET PUT POST DELETE	リクエスト先のリソースで処理ができませんでした。

ステータスコード		メソッド	説明
404	NOT_FOUND	すべて	URI が示すリソースが存在しません。
405	METHOD_NOT_ALLOWED	GET PUT POST DELETE	リクエストされた処理は、URI が示すリソースに対して有効ではありません。
406	NOT_ACCEPTABLE	GET	リクエスト時に指定された Accept ヘッダーに、不正な MIME タイプが指定されています。
409	CONFLICT	PUT POST	PUT メソッドをリクエストした場合、リソースがすでに存在していたため、HVFP はリソースを作成できませんでした。 Routing リソースに対して POST メソッドをリクエストした場合、ルートがすでに存在していたため、HVFP は新しいルートを設定できませんでした。
415	UNSUPPORTED_MEDIA_TYPE	すべて	HVFP は、Content-Type または Accept ヘッダーで指定された MIME タイプをサポートしていません。
500	INTERNAL_SERVER_ERROR	すべて	リクエスト先のノードの処理でエラーが発生しました。
503	SERVICE_UNAVAILABLE	すべて	システムに負荷が掛かっている、またはメンテナンス中などの状態にあるため、一時的にリクエストを受け付けることができません。 しばらくしてから再度リクエストしてください。 システムまたはリソースが許容できないリクエストの場合や、処理が 30 分を超えてタイムアウトした場合も、このステータスコードが返されます。

## 1.8 入出力形式

API を使用してリソースを作成または変更する場合、XML 形式でリソースのプロパティを指定します。リソースの情報を取得する場合、OPTIONS リクエストに対するレスポンスは WADL で返されます。

HVFP の API がリクエストボディとレスポンスボディに使用する XML スキーマは、次の URI からダウンロードできます。

`https://<HVFPのノードのホスト名またはIPアドレス>:9090/mapi/static/mapi.xsd`

API を通じて返されるすべてのレスポンスは、UTF-8 でエンコードされます。HVFP の API に入力するために作成するリクエストボディも UTF-8 でエンコードする必要があります。

HVFP の API は、次の表に示す MIME タイプをサポートしています。これらの MIME タイプは、リクエストおよびレスポンスでの入出力形式として指定できます。

表 1-4: サポートする MIME タイプ

MIME タイプ	説明
application/xml	XML 形式
application/octet-stream	複数のリクエスト（ファイル内容のアップロードおよびダウンロードの場合）

## 1.9 クエリパラメーター

HVFP の API では、次の表に示すクエリパラメーターを使用できます。



クエリパラメーターは、リソース URI のあとの疑問符 (?) に続けて指定します。複数のクエリパラメーターを指定する場合は、アンパサンド (&) でつなぎます。クエリパラメーターは、大文字と小文字が区別されます。

表 1-5: サポートするクエリパラメーター

クエリパラメーター	使用できるリソース	使用できるメソッド	説明
prettyprint	すべて	すべて	出力形式を指定するパラメーターです。 このパラメーターを指定すると、改行やインデントが入った見やすい形式で XML が出力されます。 このパラメーターを省略した場合は、改行やインデントが入っていない XML が出力されます。 指定例： FileSystems?prettyprint
SHARE-DIRECTORY-NAME	CIFSShares/ <Base64 でエンコードされた CIFS 共有名> NFSShares/ <NFS 共有ディレクトリ名>	GET POST DELETE HEAD	共有ディレクトリを指定するパラメーターです。 CIFSShares/ <Base64 でエンコードされた CIFS 共有名> リソースまたは NFSShares/ <NFS 共有ディレクトリ名> リソースを使用する場合は必ず指定してください。 指定例： CIFSShares/ <Base64 でエンコードされた CIFS 共有名> ?SHARE-DIRECTORY-NAME= <Base64 でエンコードされた共有ディレクトリのパス名> NFSShares/ <NFS 共有ディレクトリ名> ?SHARE-DIRECTORY-NAME= <共有ディレクトリのパス名>
verbose	すべて	GET	リクエストに対する出力内容を指定するパラメーターです。 true を指定すると、リソースに含まれるすべてのプロパティが出力されます。 false を指定すると、PUT または POST メソッドが処理できるプロパティだけが出力されます。 出力された XML は、PUT または POST メソッドをリクエストするときの XML テンプレートとして使用できます。 このパラメーターを省略した場合は、false が指定されたときと同じ動作をします。 指定例： ProcessingNode?verbose=true

## 1.10 リクエストヘッダー

HVFP へ各メソッドをリクエストするときは、リクエストヘッダーを送信します。API クライアントが設定する必要があるリクエストヘッダーの内容を次の表に示します。

表 1-6: リクエストヘッダーの内容

リクエストヘッダー	説明
Cookie	リクエストについてのユーザー認証の詳細です。 リクエスト中に api-auth Cookie がない場合、エラーが返されます。api-auth Cookie については、「 <a href="#">2.3</a> 」を参照してください。

リクエストヘッダー	説明
Accept	HVFP API クライアントが処理できる、レスポンス内容の MIME タイプの一覧です。 無効な MIME タイプが設定されている場合、エラーが返されます。指定できる MIME タイプについては、「表 1-4:」を参照してください。
Content-Type	受信側に送られる内容のメディアタイプです。 PUT または POST メソッドをリクエストする場合に使用できます。 無効な MIME タイプが設定されている場合、エラーが返されます。指定できる MIME タイプについては、「表 1-4:」を参照してください。

## 1.11 レスポンスヘッダー

HVFP の API が設定するレスポンスヘッダーの内容を次の表に示します。

表 1-7: レスポンスヘッダーの内容

レスポンスヘッダー	説明	設定される値の形式	設定条件
Status Code	処理結果を示すステータスコードです。 ステータスコードは、すべてのタイプのリクエストに設定されます。	数字	常に設定されます。
X-NODE-APIVersionID	HVFP の API のバージョン番号です。 対応する CLI コマンドまたはプロパティが存在しない場合は、Unknown が表示されます。	文字列 (HVFP のバージョン番号と同じ形式)	常に設定されます。
Content-Type	受信側に送られる内容のメディアタイプです。	MIME タイプ application/xml または text/plain が返されます。application/xml が返された場合は、ヘッダーだけでなく、ボディ部分の内容も確認してください。	application/xml は、GET メソッドのリクエストに成功した場合に設定されます。
Content-Length	受信側に送られる内容のサイズです。オクテット (8 ビット) の 10 進数で示されます。	数字	GET メソッドのリクエストに成功した場合に設定されます。
X-NODE-ErrorMessageID	処理中に発生したエラーに関するメッセージの ID です。	文字列	リクエストが失敗した場合に設定されます。
X-NODE-OperationList	リクエストで実行されたすべての処理の名称です。 リクエストが実行されなかった場合は、ハイフン (-) が表示されます。	文字列 次の形式で表示されます。 [1] <処理の名称 1> , [2] <処理の名称 2> , ... 複数の処理が実行される場合、コンマ (,) で区切られます。	常に設定されます。

レスポンスヘッダー	説明	設定される値の形式	設定条件
X-NODE-CompletedOperation	成功した処理の数です。 リクエストが実行されなかった場合は、ハイフン(-)が表示されます。	数字	常に設定されます。

## 1.12 エラー発生時のレスポンス XML

処理中に発生したエラーの内容は、API を通じてレスポンス XML として返されます。エラーが発生した場合のレスポンス XML の構造を次の表に示します。

表 1-8：エラー発生時のレスポンス XML の構造

プロパティ		説明
Error		--
--	messageId	メッセージのメッセージ ID が表示されます。
	message	メッセージの本文が表示されます。
	cause	エラーの原因が表示されます。
	recoveryAction	エラーへの対処方法が表示されます。



## アクセスと認証

HVFP の API では、リソースは URI で表されます。各リクエストでは、URI を一つ指定し、API で HVFP にアクセスするときに使用するアカウントの証明書を含む必要があります。

この章では、HVFP にアクセスするための URI、API のリクエストに指定する URI についての考慮事項、認証情報の設定方法、および SSL 証明書について説明します。

- [2.1 HVFP にアクセスするための URI とアカウント](#)
- [2.2 URI についての考慮事項](#)
- [2.3 認証](#)
- [2.4 公開鍵証明書](#)

# 2.1 HVFP にアクセスするための URI とアカウント

HVFP の API では、リクエストに次のリソース URI を指定します。

```
https://<HVFPのノードのホスト名またはIPアドレス>:9090/mapi
```

URI には、常にポート番号 9090 が含まれている必要があります。9090 は、HVFP の API が使用するポートです。mapi は、大文字と小文字が区別されますので注意してください。

API にアクセスするときに使用するアカウントは admin です。

クラスタ構成の場合、admin の初期パスワードは chang3me! です。不正なアクセスを防止するため、パスワードを必ず変更してください。admin のパスワードを変更する方法については、「[3.2 APIAccount](#)」を参照してください。

# 2.2 URI についての考慮事項

API のリクエストに指定する URI についての考慮事項を説明します。

## 2.2.1 URI の文字長

mapi のあとに続く URI の部分（追加されたクエリパラメーターを除いた部分）は、4,095 バイトに制限されます。制限を超える長さの URI がリクエストに含まれていた場合、HVFP はステータスコード 414 (REQUEST\_URI\_TOO\_LARGE) を返します。

## 2.2.2 特定の文字または値のエンコーディング

URI 中に使用する場合に、エンコーディングが必要な文字と値があります。

次の値は、Base64 でエンコードしてください。

- ・ < CIFS 共有名 >
- ・ < ポリシー名 >
- ・ クエリパラメーター SHARE-DIRECTORY-NAME

また、パーセントエンコードが必要な文字があります。パーセントエンコードする文字を次の表に示します。

表 2-1：パーセントエンコードする文字

文字	パーセントエンコードする値
スペース	%20
タブ	%09
改行	%0A
復帰	%0D
正符号 (+)	%2B
パーセント (%)	%25
番号記号 (#)	%23
疑問符 (?)	%3F
アンパサンド (&)	%26
斜線 (/) *	%2F

注：Base64 エンコーディングの結果に上記の文字が含まれている場合は、パーセントエンコードが必要です。ただし、URI の区切り文字として使用する場合は、パーセントエンコードしないでください。

エンコーディングの例を次に示します。

1. CIFS 共有名を Base64 でエンコードします。  
Base64 エンコーディングの結果が PX5+U2hhcmU= であるとしします。
2. PX5+U2hhcmU= の正符号 (+) をパーセントエンコードして %2B に置き換えます。  
パーセントエンコーディングの結果は、PX5%2BU2hhcmU= になります。

### 2.2.3 コマンドラインでの引用符の使用

Windows または UNIX のシェルが URI 中の特別な文字を誤って解釈するのを防ぐために、必ず URI 全体を引用符 (") で囲んでください。

### 2.2.4 マルチバイト文字の使用

次のリソースのリクエスト URI およびリクエスト XML には、Unicode (UTF-8) のマルチバイト文字を使用できます。

- /CIFSShares/ < Base64 でエンコードされた CIFS 共有名 >

## 2.3 認証

API で HVFP にアクセスするためには、ユーザー名とパスワードで構成される認証情報が必要になります。認証情報はリクエストごとに必要です。認証情報を設定するには、api-auth Cookie を使用します。

リクエストは、処理される前に必ず認証されます。

api-auth Cookie の形式を次に示します。

api-auth=<Base64でエンコードされたユーザー名>:<MD5ハッシュアルゴリズムで生成されたパスワード>

api-auth Cookie の値は、プレーンテキストを Base64 エンコード値および MD5 ハッシュ値に変換するツールを使用して生成してください。ただし、対話的な Web ベースのツールは、安全性が低い使用しないことを推奨します。生成した api-auth Cookie を、リクエストヘッダーに指定します。

リクエストヘッダーで送られた認証情報は、データベースに格納されているユーザー情報と照合されます。サーバに送られたリクエストの認証が成功すると、リクエストが有効になり、形式が正しいかチェックされます。認証に失敗した場合は、処理が終了し、異常終了した場合のステータスコードが返されます。

注意：

パスワードを忘れた場合は認証できません。その場合は、adminpasswd コマンドでパスワードを初期化したあと、パスワードを変更してください。

## 2.4 公開鍵証明書

HVFP に格納されている公開鍵証明書は、自己署名型の証明書です。certctl コマンドを使用することで、認証局 (CA) から取得した公開鍵証明書をノードまたは Virtual Server に設定できま

す。公開鍵証明書を設定する手順については、「ユーザーズガイド」(IF205A)を参照してください。

API クライアントには、HVFP から送られたサーバ証明書を信頼できるサーバ証明書としてインストールしてください。サーバ証明書としてインストールできない場合は、サーバ認証の手続きを省略して、SSL による暗号化通信に使用してください。



## リソース

この章では、それぞれのリソースで利用できる HTTP メソッドの種類、各メソッドのリクエスト XML とレスポンス XML の構造、プロパティの詳細、リクエスト先の URI、およびメソッドが終了したときに返されるステータスコードについて説明します。なお、すべてのリソースに対して OPTIONS メソッドを使用できますが、この章の表では省略します。

- ❑ [3.1 リソースの一覧](#)
- ❑ [3.2 APIAccount](#)
- ❑ [3.3 CIFSShare](#)
- ❑ [3.4 FCPath](#)
- ❑ [3.5 FileSystem](#)
- ❑ [3.6 Hardware](#)
- ❑ [3.7 LU](#)
- ❑ [3.8 NetworkInterface](#)
- ❑ [3.9 NFSShare](#)
- ❑ [3.10 ProcessingNode](#)
- ❑ [3.11 Routing](#)
- ❑ [3.12 SystemController](#)
- ❑ [3.13 VirtualServer](#)

## 3.1 リソースの一覧

リソースの使用目的と参照先を次の表に示します。

表 3-1：リソースの使用目的

使用目的	リソース名	説明	参照先
ユーザーアカウントの管理	APIAccount	API の管理者アカウントの管理	<a href="#">3.2</a>
ノードの管理	ProcessingNode	クラスタおよびノードの情報の管理	<a href="#">3.10</a>
ネットワークの管理	NetworkInterface	ネットワークインターフェースの情報の管理	<a href="#">3.8</a>
	Routing	ルーティング情報の管理	<a href="#">3.11</a>
サービスの管理	SystemController	サービスおよび OS の制御	<a href="#">3.12</a>
状態の取得	FCPath	FC パスの状態の取得	<a href="#">3.4</a>
	Hardware	ハードウェアの状態の取得	<a href="#">3.6</a>
	VirtualServer	Virtual Server の状態の取得	<a href="#">3.13</a>
ファイルシステムの管理	LU	LU の状態の取得	<a href="#">3.7</a>
	FileSystem	ファイルシステムの情報の管理	<a href="#">3.5</a>
	VolumeGroup	ボリュームグループの情報の取得	
共有の管理	CIFSShare	CIFS 共有の情報の管理	<a href="#">3.3</a>
	NFSShare	NFS 共有の情報の管理	<a href="#">3.9</a>

## 3.2 APIAccount

### 3.2.1 APIAccount リソースの概要

APIAccount リソースを使用して、API の管理者アカウントのパスワードを変更したり、情報を取得したりできます。

APIAccount リソースに使用できる HTTP メソッドを次の表に示します。

表 3-2：APIAccount リソースに使用できる HTTP メソッド

リソースの URI	HTTP メソッド	サポート構成	説明
/APIAccount	GET	クラスタ Virtual Server	API の管理者アカウントの情報を取得します。
	POST	クラスタ Virtual Server	API の管理者アカウントのパスワードを変更します。

### 3.2.2 APIAccount リソース

この項では、APIAccount リソースの使い方について説明します。

#### 3.2.2.1 GET メソッドで API の管理者アカウントの情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

表 3-3 : APIAccount リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ		出力される個数	サポート構成	verbose が false の場合の表示
APIAccount		1	クラスタ Virtual Server	○
--	userName	1	クラスタ Virtual Server	○
	password	1	クラスタ Virtual Server	○ (空文字列)

(凡例) ○ : 表示されます。

APIAccount リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

表 3-4 : APIAccount リソースに GET メソッドをリクエストしたときのプロパティ

プロパティ		説明
APIAccount		--
--	userName	API の管理者アカウントのユーザー名 (admin) が表示されます。
	password	常に空文字列が表示されます。

APIAccount リソースに GET メソッドをリクエストして、API の管理者アカウントの情報を取得する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/APIAccount

#### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<APIAccount>
  <userName>admin</userName>
  <password></password>
</APIAccount>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-5 : APIAccount リソースに GET メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.2.2.2

#### POST メソッドで API の管理者アカウントのパスワードを変更する

クラスタ構成の場合、初期パスワードは chang3me! です。不正なアクセスを防止するため、パスワードを必ず変更してください。

重要 :

設定したパスワードは、HVFP を運用および管理するために必要です。忘れないようにしてください。

注意 :

リクエストは、クラスタを構成しているノードのうち、実行したノードだけで有効になります。クラスタ構成の場合でパスワードを変更するときは、両方のノードに同じ POST メソッドをリクエストしてください。

APIAccount リソースに POST メソッドをリクエストするときの XML の構造を次の表に示します。

**表 3-6 : APIAccount リソースに POST メソッドをリクエストするときの XML の構造**

プロパティ		データ型	指定できる個数	サポート構成
APIAccount		--	1	クラスタ Virtual Server
--	userName	String	1	クラスタ Virtual Server
	password	String	1	クラスタ Virtual Server

APIAccount リソースに POST メソッドをリクエストするときに指定するプロパティについて次の表に示します。

**表 3-7 : APIAccount リソースに POST メソッドをリクエストするときのプロパティ**

プロパティ		説明	指定時の条件
APIAccount		--	--
--	userName	API の管理者のユーザー名を設定します。設定できる文字列は、admin だけです。	必須
	password	API の管理者のパスワードを設定します。使用できるのは、英数字、感嘆符 (!)、番号記号 (#)、ドル記号 (\$)、パーセント (%), アンパサンド (&)、アポストロフィ (')、始め丸括弧 ((), 終わり丸括弧 ()), アステリスク (*), 正符号 (+), ハイフン (-), ピリオド (.), 等号 (=), アクサンシルコンフレックス (^), アンダーライン (_), 縦線 ( ), 円記号 (¥), 単価記号 (@) およびスペースです。ただし、スペースはパスワードの先頭および末尾には使用できません。 指定できる文字列の長さは、4 ～ 256 文字です。	必須

APIAccount リソースに POST メソッドをリクエストして、API の管理者アカウントのパスワードを変更する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/APIAccount

#### リクエスト XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<APIAccount>
  <userName>admin</userName>
  <password>password</password>
</APIAccount>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-8 : APIAccount リソースに POST メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
400	メソッドの実行に失敗しました。
500	SQL の例外、またはデータベースの接続エラーが発生しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

## 3.3 CIFSShare

### 3.3.1 CIFSShare リソースの概要

CIFSShare リソースを使用して、CIFS 共有を管理できます。

CIFSShare リソースに使用できる HTTP メソッドを次の表に示します。

表 3-9：CIFSShare リソースに使用できる HTTP メソッド

リソースの URI	HTTP メソッド	サポート構成	説明
/CIFSShares	GET	クラスタ Virtual Server	ログインしているノード上または VirtualServer 上の CIFS 共有の情報を取得します。
	PUT	クラスタ Virtual Server	CIFS 共有を追加します。
/CIFSShares/<Base64 でエンコードされた CIFS 共有名>	GET*	クラスタ Virtual Server	指定した CIFS 共有の情報を取得します。
	POST*	クラスタ Virtual Server	指定した CIFS 共有の情報を変更します。
	DELETE*	クラスタ Virtual Server	指定した CIFS 共有を削除します。
	HEAD*	クラスタ Virtual Server	指定した CIFS 共有が存在するかどうかを確認します。

注\* クエリパラメーター SHARE-DIRECTORY-NAME を使用して、Base64 でエンコードされた共有ディレクトリのパス名を必ず指定してください。

### 3.3.2 CIFSShare リソース

この項では、CIFSShare リソースの使い方について説明します。

#### 3.3.2.1 GET メソッドですべての CIFS 共有の情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

表 3-10：CIFSShare リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ		出力される 個数	サポート構成	verbose が false の場合の表示
CIFSShares		1	クラスタ Virtual Server	○
--	CIFSShare	0, 1 ~ n	クラスタ Virtual Server	○
	-- exportPoint	1	クラスタ Virtual Server	○
	cifsShareName	1	クラスタ Virtual Server	○
	isAclEnabled	1	クラスタ Virtual Server	○
	restrictionTargetHosts	1	クラスタ Virtual Server	○
	isAccessRestrictionAllowed	1	クラスタ Virtual Server	○

プロパティ		出力される 個数	サポート構成	verbose が false の場合の表示
	commentForShare	1	クラスタ Virtual Server	○
	isReadOnly	1	クラスタ Virtual Server	○
	isBrowseEnabled	1	クラスタ Virtual Server	○
	fileAccessPermissionOfOwner	1	クラスタ Virtual Server	○
	fileAccessPermissionOfGroup	1	クラスタ Virtual Server	○
	fileAccessPermissionOfOther	1	クラスタ Virtual Server	○
	directoryAccessPermissionOfOwner	1	クラスタ Virtual Server	○
	directoryAccessPermissionOfGroup	1	クラスタ Virtual Server	○
	directoryAccessPermissionOfOther	1	クラスタ Virtual Server	○
	guestAccessPermission	1	クラスタ Virtual Server	○
	diskSynchronizationPolicy	1	クラスタ Virtual Server	○
	cifsClientCacheSetting	1	クラスタ Virtual Server	○
	fileTimestampChangeableUsers	1	クラスタ Virtual Server	○
	isHomeDirectoryEnabled	1	クラスタ Virtual Server	○
	aclType	1	クラスタ Virtual Server	×
	clientAccessPolicy	1	クラスタ Virtual Server	○
	vssUse	1	クラスタ Virtual Server	○
	cifsClientRoCacheOptimize	1	クラスタ Virtual Server	○
	accessBasedEnumeration	1	クラスタ Virtual Server	○
	writeDisallowedUsers	1	クラスタ Virtual Server	○
	writeDisallowedGroups	1	クラスタ Virtual Server	○
	writeAllowedUsers	1	クラスタ Virtual Server	○
	writeAllowedGroups	1	クラスタ Virtual Server	○

(凡例) n : 2 以上の整数を示します。○ : 表示されます。× : 表示されません。

CIFSShare リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

表 3-11 : CIFSShare リソースに GET メソッドをリクエストしたときのプロパティ

プロパティ		説明
CIFSShares		--
--	CIFSShare	--
	exportPoint	共有ディレクトリが絶対パスで表示されます。
	cifsShareName	CIFS 共有が作成された CIFS 共有名が表示されます。
	isAclEnabled	CIFS クライアントに ACL の参照・設定を許可しているかどうかが表示されます。 true CIFS クライアントに ACL の参照・設定を許可している場合に表示されます。 false CIFS クライアントに ACL の参照・設定を許可していない場合に表示されます。
	restrictionTargetHosts	アクセス制限の設定対象となっているホストが表示されます。
	isAccessRestrictionAllowed	アクセス制限が設定されているかどうかが表示されます。 true アクセス制限が設定されている場合に表示されます。 false アクセス制限が設定されていない場合に表示されます。
	commentForShare	CIFS 共有のコメントが表示されます。
	isReadOnly	CIFS 共有のアクセス権が表示されます。 true 読み取り専用設定されている場合に表示されます。 false 読み取りおよび書き込みが許可されている場合に表示されます。
	isBrowseEnabled	クライアント環境で CIFS 共有名を一覧に表示することが許可されているかどうかが表示されます。 true 許可されている場合に表示されます。 false 許可されていない場合に表示されます。

プロパティ		説明
	fileAccessPermissionOfOwner	<p>書き込みが許可された CIFS 共有にファイルを作成するときのオーナーのアクセス権が表示されます。</p> <p>rw 読み取りおよび書き込みが許可されている場合に表示されます。</p> <p>ro 読み取りだけが許可されている場合に表示されます。</p> <p>none 読み取りおよび書き込みが許可されていない場合に表示されます。</p> <p>Unknown ACL タイプが取得できなかった場合に表示されます。</p> <p>オーナーだけにアクセス権を指定した場合は、空文字列が表示されます。</p>
	fileAccessPermissionOfGroup	<p>書き込みが許可された CIFS 共有にファイルを作成するときのグループのアクセス権が表示されます。</p> <p>表示される値の詳細については、fileAccessPermissionOfOwner プロパティを参照してください。</p>
	fileAccessPermissionOfOther	<p>書き込みが許可された CIFS 共有にファイルを作成するときのその他のアクセス権が表示されます。</p> <p>表示される値の詳細については、fileAccessPermissionOfOwner プロパティを参照してください。</p>
	directoryAccessPermissionOfOwner	<p>書き込みが許可された CIFS 共有にディレクトリを作成するときのオーナーのアクセス権が表示されます。</p> <p>表示される値の詳細については、fileAccessPermissionOfOwner プロパティを参照してください。</p>
	directoryAccessPermissionOfGroup	<p>書き込みが許可された CIFS 共有にディレクトリを作成するときのグループのアクセス権が表示されます。</p> <p>表示される値の詳細については、fileAccessPermissionOfOwner プロパティを参照してください。</p>
	directoryAccessPermissionOfOther	<p>書き込みが許可された CIFS 共有にディレクトリを作成するときのその他のアクセス権が表示されます。</p> <p>表示される値の詳細については、fileAccessPermissionOfOwner プロパティを参照してください。</p>



プロパティ			説明
		guestAccessPermission	<p>ゲストアカウントのアクセスが許可されているかどうかが表示されます。</p> <p>allow CIFS 共有にゲストアカウントでのアクセスが許可されている場合に表示されます。</p> <p>disallow CIFS 共有にゲストアカウントでのアクセスが許可されていない場合に表示されます。</p> <p>default GUI の [CIFS Service Management] ページ (Setting Type : Security) で設定した CIFS サービス構成定義が適用されている場合に表示されます。</p>
		diskSynchronizationPolicy	<p>CIFS クライアントから CIFS 共有への書き込み要求に対する動作の設定が表示されます。</p> <p>write_and_close 書き込みおよびクローズ要求に同期して書き込む場合に表示されます。</p> <p>close クローズ要求に同期して書き込む場合に表示されます。</p> <p>none 一定周期で書き込む設定の場合に表示されます。</p> <p>default GUI の [CIFS Service Management] ページ (Setting Type : Performance) で設定した CIFS サービス構成定義が適用されている場合に表示されます。</p>
		cifsClientCacheSetting	<p>CIFS 共有内のファイルの更新データをクライアントにキャッシュするかどうかの設定が表示されます。</p> <p>use クライアントにキャッシュする設定の場合に表示されます。</p> <p>do_not_use クライアントにキャッシュしない設定の場合に表示されます。</p> <p>default GUI の [CIFS Service Management] ページ (Setting Type : Performance) で設定した CIFS サービス構成定義が適用されている場合に表示されます。</p>

プロパティ		説明
	fileTimestampChangeableUsers	<p>CIFS 共有内のファイルのタイムスタンプを更新できるユーザーが表示されます。</p> <p>write_permitted_users ファイルの書き込みを許可されたすべてのユーザーが、タイムスタンプを更新できる場合に表示されます。</p> <p>owner_only ファイルのオーナーだけがタイムスタンプを更新できる場合に表示されます。</p> <p>Unknown ACL タイプを取得できなかった場合に表示されます。</p> <p>default GUI の [CIFS Service Management] ページ (Setting Type : Security) で設定した CIFS サービス構成定義が適用されている場合に表示されます。 ファイルシステムが Advanced ACL の場合は、空文字列が表示されます。</p>
	isHomeDirectoryEnabled	<p>CIFS 共有でホームディレクトリの自動作成機能が有効かどうかが表示されます。</p> <p>true ホームディレクトリの自動作成機能が有効な場合に表示されます。</p> <p>false ホームディレクトリの自動作成機能が無効な場合に表示されます。</p>
	aclType	<p>CIFS 共有が作成されたファイルシステムの ACL タイプが表示されます。</p> <p>Advanced ACL Advanced ACL の場合に表示されます。</p> <p>Classic ACL Classic ACL の場合に表示されます。</p> <p>Unknown ACL タイプを取得できなかった場合に表示されます。</p>
	clientAccessPolicy	<p>CIFS クライアントからのアクセスの処理方法が表示されます。</p> <p>parallel アクセスが平行に処理される場合に表示されます。</p> <p>serial アクセスがシリアルに処理される場合に表示されます。</p> <p>default GUI の [CIFS Service Management] ページ (Setting Type : Performance) で設定した CIFS サービス構成定義が適用されている場合に表示されます。</p>

プロパティ		説明
	vssUse	<p>Volume Shadow Copy Service を使用しているかどうかが表示されます。</p> <p>default GUI の [CIFS Service Management] ページ (Setting Type : Basic) で設定した CIFS サービス構成定義が適用されている場合に表示されます。</p> <p>do_not_use Volume Shadow Copy Service を使用していない場合に表示されます。</p>
	cifsClientRoCacheOptimize	<p>複数の CIFS クライアントによるファイルへのアクセス競合が発生した場合に、読み取り専用のクライアントキャッシュを使用しているかどうかが表示されます。</p> <p>use 読み取り専用のクライアントキャッシュを使用している場合に表示されます。</p> <p>do_not_use 読み取り専用のクライアントキャッシュを使用していない場合に表示されます。</p> <p>default GUI の [CIFS Service Management] ページ (Setting Type : Performance) で設定した CIFS サービス構成定義が適用されている場合に表示されます。</p>
	accessBasedEnumeration	<p>アクセスベースの列挙が有効かどうかが表示されます。</p> <p>use アクセスベースの列挙が有効な場合に表示されます。</p> <p>do_not_use アクセスベースの列挙が無効な場合に表示されます。</p> <p>default GUI の [CIFS Service Management] ページ (Setting Type : Security) で設定した CIFS サービス構成定義が適用されている場合に表示されます。</p>
	writeDisallowedUsers	<p>CIFS 共有への書き込みを拒否するユーザーが表示されます。</p> <p>書き込みを拒否するユーザーが設定されていない場合は、空文字列が表示されます。</p>
	writeDisallowedGroups	<p>CIFS 共有への書き込みを拒否するグループが表示されます。</p> <p>書き込みを拒否するグループが設定されていない場合は、空文字列が表示されます。</p>
	writeAllowedUsers	<p>CIFS 共有への書き込みを許可するユーザーが表示されます。</p> <p>書き込みを許可するユーザーが設定されていない場合は、空文字列が表示されます。</p>
	writeAllowedGroups	<p>CIFS 共有への書き込みを許可するグループが表示されます。</p> <p>書き込みを許可するグループが設定されていない場合は、空文字列が表示されます。</p>

CIFSShare リソースに GET メソッドをリクエストして、すべての CIFS 共有の情報を取得する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/CIFSShares

### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CIFSShares>
  <CIFSShare>
    <exportPoint>/mnt/aaa/sh1</exportPoint>
    <cifsShareName>sh</cifsShareName>
    <isAclEnabled>true</isAclEnabled>
    <restrictionTargetHosts>0.0.0.0</restrictionTargetHosts>
    <isAccessRestrictionAllowed>>false</isAccessRestrictionAllowed>
    <commentForShare></commentForShare>
    <isReadOnly>>false</isReadOnly>
    <isBrowseEnabled>true</isBrowseEnabled>
    <fileAccessPermissionOfOwner></fileAccessPermissionOfOwner>
    <fileAccessPermissionOfGroup></fileAccessPermissionOfGroup>
    <fileAccessPermissionOfOther></fileAccessPermissionOfOther>
    <directoryAccessPermissionOfOwner></
directoryAccessPermissionOfOwner>
    <directoryAccessPermissionOfGroup></
directoryAccessPermissionOfGroup>
    <directoryAccessPermissionOfOther></
directoryAccessPermissionOfOther>
    <guestAccessPermission>default</guestAccessPermission>
    <diskSynchronizationPolicy>default</diskSynchronizationPolicy>
    <cifsClientCacheSetting>default</cifsClientCacheSetting>
    <fileTimestampChangeableUsers></fileTimestampChangeableUsers>
    <isHomeDirectoryEnabled>>false</isHomeDirectoryEnabled>
    <clientAccessPolicy>default</clientAccessPolicy>
    <vssUse>default</vssUse>
    <cifsClientRoCacheOptimize>default</cifsClientRoCacheOptimize>
    <accessBasedEnumeration>default</accessBasedEnumeration>
    <writeDisallowedUsers></writeDisallowedUsers>
    <writeDisallowedGroups></writeDisallowedGroups>
    <writeAllowedUsers></writeAllowedUsers>
    <writeAllowedGroups></writeAllowedGroups>
  </CIFSShare>
</CIFSShares>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-12: CIFSShare リソースに GET メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3: 」を参照してください。

### 3.3.2.2 PUT メソッドで CIFS 共有を追加する

CIFSShare リソースに PUT メソッドをリクエストするときの XML の構造を次の表に示します。

表 3-13: CIFSShare リソースに PUT メソッドをリクエストするときの XML の構造

プロパティ	データ型	指定できる個数	サポート構成
CIFSShare	--	1	クラスタ Virtual Server

プロパティ		データ型	指定できる個数	サポート構成
--	exportPoint	String	1	クラスタ Virtual Server
	cifsShareName	String	1	クラスタ Virtual Server
	isClassicAcl	Boolean	1	クラスタ Virtual Server
	isAclEnabled	Boolean	0, 1	クラスタ Virtual Server
	restrictionTargetHosts	String	0, 1	クラスタ Virtual Server
	isAccessRestrictionAllowed	Boolean	0, 1	クラスタ Virtual Server
	commentForShare	String	0, 1	クラスタ Virtual Server
	isReadOnly	Boolean	0, 1	クラスタ Virtual Server
	isBrowseEnabled	Boolean	0, 1	クラスタ Virtual Server
	fileAccessPermissionOfOwner	String	0, 1	クラスタ Virtual Server
	fileAccessPermissionOfGroup	String	0, 1	クラスタ Virtual Server

プロパティ		データ型	指定できる個数	サポート構成	
	fileAccessPermissionOfOther	String	0, 1	クラスタ Virtual Server	
	directoryAccessPermissionOfOwner	String	0, 1	クラスタ Virtual Server	
	directoryAccessPermissionOfGroup	String	0, 1	クラスタ Virtual Server	
	directoryAccessPermissionOfOther	String	0, 1	クラスタ Virtual Server	
	guestAccessPermission	String	0, 1	クラスタ Virtual Server	
	diskSynchronizationPolicy	String	0, 1	クラスタ Virtual Server	
	cifsClientCacheSetting	String	0, 1	クラスタ Virtual Server	
	fileTimestampChangeableUsers	String	0, 1	クラスタ Virtual Server	
	isHomeDirectoryEnabled	Boolean	0, 1	クラスタ Virtual Server	
	clientAccessPolicy	String	0, 1	クラスタ Virtual Server	
	vssUse	String	0, 1	クラスタ Virtual Server	
	cifsClientRoCacheOptimize	String	0, 1	クラスタ Virtual Server	
	accessBasedEnumeration	String	0, 1	クラスタ Virtual Server	
	writeDisallowedUsers	String	0, 1	クラスタ Virtual Server	
	writeDisallowedGroups	String	0, 1	クラスタ Virtual Server	
	writeAllowedUsers	String	0, 1	クラスタ Virtual Server	
	writeAllowedGroups	String	0, 1	クラスタ Virtual Server	
	DirectorySetup	--	0, 1	クラスタ Virtual Server	
	--	userName	String	0, 1	クラスタ Virtual Server
		groupName	String	0, 1	クラスタ Virtual Server
		isStickyBit	Boolean	0, 1	クラスタ Virtual Server
		ownerPermission	String	0, 1	クラスタ Virtual Server
		groupPermission	String	0, 1	クラスタ Virtual Server
		otherPermission	String	0, 1	クラスタ Virtual Server

プロパティ				データ型	指定できる個数	サポート構成
	DirectoryACLSettings			--	0, 1	クラスタ Virtual Server
	--	DirectoryACLSetting		--	0, 1 ~ n	クラスタ Virtual Server
		--	userGroupAccess	String	1	クラスタ Virtual Server
			name	String	1	クラスタ Virtual Server
			accountType	String	1	クラスタ Virtual Server
			applicationDestination	String	1	クラスタ Virtual Server
			inheritanceRange	String	1	クラスタ Virtual Server
			accessPermission	String	1	クラスタ Virtual Server

(凡例) n : 2 以上の整数を示します。

CIFSShare リソースに PUT メソッドをリクエストするときに指定するプロパティについて次の表に示します。

表 3-14 : CIFSShare リソースに PUT メソッドをリクエストするときのプロパティ

プロパティ	説明	指定時の条件
CIFSShare	--	--

プロパティ		説明	指定時の条件
--	exportPoint	<p>CIFS 共有を作成するディレクトリを 6 文字以上 256 文字以内の絶対パスで指定します。差分スナップショットの自動作成スケジュール（共有の自動作成）を運用する場合、絶対パスが 241 文字以内になるようにしてください。</p> <p>このプロパティに「/mnt/&lt;ファイルシステム名&gt;」で始まるパスを指定することで、ファイルシステムと CIFS 共有が対応します。</p> <p>例えば、filesystem01 ファイルシステムの share01 ディレクトリにファイル共有を作成する場合は、/mnt/filesystem01/share01 と指定します。</p> <p>指定できる文字は英数字、感嘆符 (!)、番号記号 (#)、ドル記号 (\$)、パーセント (%), アンバサンド (&amp;), アポストロフィ ('), 始め丸括弧 ((), 終わり丸括弧 ()), 正符号 (+), コンマ (,), ハイフン (-), ピリオド (.), 斜線 (/), セミコロン (;), 等号 (=), 単価記号 (@), 始め角括弧 ([), 終わり角括弧 (]), アクサンシルコンフレックス (^), アンダーライン (_), アクサングラフ (ˆ), 始め波括弧 ({), 終わり波括弧 (}), 波ダッシュ (~) およびスペースです。</p> <p>このほか、マルチバイト文字も指定できます。</p> <p>共有ディレクトリはクラスタ内で一意である必要があります。</p> <p>Virtual Server 上でファイル共有を追加する場合は、Virtual Server 内で重複しない名称を指定します。なお、「.conflict」, 「.conflict_longpath」, 「.snaps」, 「.history」および「.lost+found」というディレクトリ名は指定できません。</p>	必須



プロパティ		説明	指定時の条件
		<p>また、「.arc」、「.system_gi」、 「.system_reorganize」、 「.backupdates」、 「.temp_backupdates」および 「lost+found」は、ファイルシス テム直下のディレクトリの名称 として指定できません。シンボ リックリンクを含むパスは指定 できません。末尾に指定した斜 線およびスペースは削除されま す。</p> <p>パーセント (%) は次の組み合 わせで共有ディレクトリ名に指 定しないでください。</p> <p>%D, %G, %H, %I, %L, %M, %N, %P, %R, %S, %T, %U, %V, %a, %d, %g, %h, %i, %m, %p, %u, %v, %w, %\$</p>	

プロパティ	説明	指定時の条件
cifsShareName	<p>ユーザーから参照できる CIFS 共有名を指定します。CIFS 共有名は、ノードで一意になるように 80 文字以内で指定します。差分スナップショットの自動作成スケジュール（共有の自動作成）を運用する場合、CIFS 共有名は 69 文字以内で指定します。</p> <p>指定できる文字は英数字、感嘆符 (!)、番号記号 (#)、ドル記号 (\$)、パーセント (%)、アンパサンド (&amp;)、アポストロフィ (')、始め丸括弧 ((), 終わり丸括弧 ()), 正符号 (+)、コンマ (,), ハイフン (-)、ピリオド (.), セミコロン (;)、等号 (=)、単価記号 (@)、始め角括弧 ([, 終わり角括弧 (]), アクサンシルコンフレックス (^)、アンダーライン (_), アクサングラーブ (`), 始め波括弧 ({, 終わり波括弧 (}), 波ダッシュ (~) およびスペースです。このほか、マルチバイト文字も指定できます。ただし、「\$」,「.」や「..」のようにドル記号 (\$) またはピリオド (.) だけを指定したり、「Abc.」や「Abc.\$」のようにピリオド (.) を文字列の末尾やドル記号 (\$) を除いた末尾に指定したりできません。</p> <p>末尾に指定したスペースは削除されます。</p> <p>パーセント (%) は次の組み合わせで CIFS 共有名に指定しないでください。</p> <p>%D, %G, %H, %I, %L, %M, %N, %P, %R, %S, %T, %U, %V, %a, %d, %g, %h, %i, %m, %p, %u, %v, %w, %x</p> <p>なお、global, homes, printers, admin\$, c\$, global\$, homes\$, ipc\$ および printers\$ は、CIFS 共有名として指定できません。入力した英大文字と英小文字は、Windows で区別されません。</p> <p>英大文字と英小文字に関係なく、一意な名称を指定してください。</p>	必須

プロパティ	説明	指定時の条件
isClassicAcl	<p>Advanced ACL または Classic ACL のどちらで操作するかを設定します。操作対象の共有ディレクトリが属するファイルシステムの ACL タイプに合わせて設定してください。</p> <p>true</p> <p>Classic ACL タイプのファイルシステムの場合に指定します。</p> <p>false</p> <p>Advanced ACL タイプのファイルシステムの場合に指定します。</p>	必須
isAclEnabled	<p>CIFS クライアントに ACL の参照・設定を許可するかどうかを指定します。</p> <p>true<sup>*1</sup></p> <p>CIFS クライアントに ACL の参照・設定を許可する場合に指定します。</p> <p>false</p> <p>CIFS クライアントに ACL の参照・設定を許可しない場合に指定します。</p>	isClassicAcl プロパティが false の場合は必ず true を指定します。
restrictionTargetHosts	<p>CIFS 共有へのアクセスを制限するホストを設定します。複数指定する場合はコンマ (,) で区切ってください。</p> <p>ホストをネットワークアドレスで指定する場合は、IP アドレスを指定します (例: 10.203.15.0)。</p> <p>ネットマスクに従ってネットワークの範囲を指定する場合は、&lt;ネットワークアドレス&gt; / &lt;ネットマスク&gt; の形式で指定します (例: 10.203.15.0 / 255.255.255.0)。</p> <p>ホスト名を指定する場合は、GUI の [Edit System File] ページで /etc/hosts ファイルを編集して、指定したすべてのホスト名と IP アドレスを追加してください。</p> <p>省略した場合、すべてのホストまたはネットワークが対象になります。</p> <p>GUI の [CIFS Service Management] ページ (Setting Type : Security) でアクセスを制限するホストまたはネットワークが設定されている場合は、すべてのファイル共有にその設定が適用されます。</p>	isAccessRestrictionAllowed プロパティを設定する場合に必須です。

プロパティ		説明	指定時の条件
	isAccessRestrictionAllowed	restrictionTargetHosts プロパティで指定したホストに対して CIFS 共有へのアクセスを制限するかどうかを設定します。 true アクセスを制限する場合に指定します。 false アクセスを制限しない場合に指定します。 GUI の [CIFS Service Management] ページ (Setting Type : Security) でアクセスを制限するホストまたはネットワークが設定されている場合は、すべてのファイル共有にその設定が適用されます。	restrictionTargetHosts プロパティを設定する場合に必須です。
	commentForShare	CIFS 共有のコメントを 256 文字以内で指定します。 設定できる文字は、英数字、感嘆符 (!)、番号記号 (#)、ドル記号 (\$)、アンパサンド (&)、アポストロフィ (')、始め丸括弧 ( ( ), 終わり丸括弧 ( ) ), アスタリスク (*), 正符号 (+), コンマ ( , ), ハイフン ( - ), ピリオド ( . ), 斜線 ( / ), コロン ( : ), 始め山括弧 ( < ), 終わり山括弧 ( > ), 疑問符 ( ? ), 単価記号 ( @ ), 始め角括弧 ( [ ), 円記号 ( ¥ ), 終わり角括弧 ( ] ), アクサンシルコンフレックス ( ^ ), アンダーライン ( _ ), アクサングラーブ ( ` ), 始め波括弧 ( { ), 縦線 (   ), 終わり波括弧 ( } ) および波ダッシュ ( ~ ) です。スペースも指定できますが、文字列の先頭および末尾には指定できません。また、円記号 ( ¥ ) は文字列の末尾には指定できません。このほか、マルチバイト文字も指定できます。 省略した場合、CIFS 共有のコメントは設定されません。	任意
	isReadOnly	CIFS 共有を読み取り専用にするかどうかを設定します。 true 読み取り専用にする場合に指定します。 false <sup>*1</sup> 読み取りと書き込みを許可する場合に指定します。	任意

プロパティ		説明	指定時の条件
	isBrowseEnabled	クライアント環境で CIFS 共有名を一覧に表示するかどうかを設定します。 true <sup>*1</sup> 表示する場合に指定します。 false 表示しない場合に指定します。	任意
	fileAccessPermissionOfOwner <sup>*2</sup>	書き込みが許可された CIFS 共有にファイルを作成するときのオーナーのアクセス権を指定します。 rw <sup>*1</sup> 読み取りおよび書き込みを許可する場合に指定します。 ro 読み取りだけを許可する場合に指定します。 none 読み取りおよび書き込みを許可しない場合に指定します。	fileAccessPermissionOfGroup プロパティおよび fileAccessPermissionOfOther プロパティが設定されている場合は必須です。
	fileAccessPermissionOfGroup <sup>*2</sup>	書き込みが許可された CIFS 共有にファイルを作成するときのグループのアクセス権を指定します。 省略した場合は、ro が設定されます。 設定できる値の詳細については、fileAccessPermissionOfOwner プロパティを参照してください。	fileAccessPermissionOfOwner プロパティおよび fileAccessPermissionOfOther プロパティが設定されている場合は必須です。
	fileAccessPermissionOfOther <sup>*2</sup>	書き込みが許可された CIFS 共有にファイルを作成するときのその他のアクセス権を設定します。 省略した場合は、ro が設定されます。 設定できる値の詳細については、fileAccessPermissionOfOwner プロパティを参照してください。	fileAccessPermissionOfOwner プロパティおよび fileAccessPermissionOfGroup プロパティが設定されている場合は必須です。
	directoryAccessPermissionOfOwner <sup>*2</sup>	書き込みが許可された CIFS 共有にディレクトリを作成するときのオーナーのアクセス権を指定します。 省略した場合は、rw が設定されます。 設定できる値の詳細については、fileAccessPermissionOfOwner プロパティを参照してください。	directoryAccessPermissionOfGroup プロパティおよび directoryAccessPermissionOfOther プロパティが設定されている場合は必須です。

プロパティ		説明	指定時の条件
	directoryAccessPermissionOfGroup <sup>*2</sup>	書き込みが許可された CIFS 共有にディレクトリを作成するときのグループのアクセス権を指定します。 省略した場合は、ro が設定されます。 設定できる値の詳細については、fileAccessPermissionOfOwner プロパティを参照してください。	directoryAccessPermissionOfOwner プロパティおよび directoryAccessPermissionOfOther プロパティが設定されている場合は必須です。
	directoryAccessPermissionOfOther <sup>*2</sup>	書き込みが許可された CIFS 共有にディレクトリを作成するときのその他のアクセス権を指定します。 省略した場合は、ro が設定されます。 設定できる値の詳細については、fileAccessPermissionOfOwner プロパティを参照してください。	directoryAccessPermissionOfOwner プロパティおよび directoryAccessPermissionOfGroup プロパティが設定されている場合は必須です。
	guestAccessPermission	ゲストアカウントのアクセスを許可するかどうかを設定します。 allow CIFS 共有にゲストアカウントでのアクセスを許可する場合に指定します。 disallow CIFS 共有にゲストアカウントでのアクセスを許可しない場合に指定します。 default <sup>*1</sup> GUI の [CIFS Service Management] ページ (Setting Type : Security) で設定した CIFS サービス構成定義に従う場合に指定します。	任意

プロパティ	説明	指定時の条件
diskSynchronizationPolicy	<p>クライアントからの書き込みまたはクローズ要求に対する動作を設定します。</p> <p>write_and_close クライアントから CIFS 共有への書き込みおよびクローズ要求に同期して書き込む場合に指定します。</p> <p>close クライアントから CIFS 共有へのクローズ要求に同期して書き込む場合に指定します。</p> <p>none 一定周期で書き込む場合に指定します。</p> <p>default*1 GUI の [CIFS Service Management] ページ (Setting Type : Performance) で設定した CIFS サービス構成定義に従う場合に指定します。</p>	任意
cifsClientCacheSetting	<p>CIFS 共有内のファイルの更新データをクライアントにキャッシュするかどうかを設定します。</p> <p>use クライアントにキャッシュする場合に指定します。</p> <p>do_not_use クライアントにキャッシュしない場合に指定します。</p> <p>default*1 GUI の [CIFS Service Management] ページ (Setting Type : Performance) で設定した CIFS サービス構成定義に従う場合に指定します。</p>	任意

プロパティ		説明	指定時の条件
	fileTimestampChangeableUsers	<p>CIFS 共有内のファイルのタイムスタンプを更新できるユーザーを設定します。</p> <p>write_permitted_users ファイルの書き込みを許可されたすべてのユーザーに、タイムスタンプの更新を許可する場合に指定します。</p> <p>owner_only ファイルのオーナーだけにタイムスタンプの更新を許可する場合に指定します。</p> <p>default<sup>*1</sup> GUI の [CIFS Service Management] ページ (Setting Type : Security) で設定した CIFS サービス構成定義に従う場合に指定します。</p>	isClassicAcl プロパティが true の場合に設定できます。
	isHomeDirectoryEnabled	<p>CIFS 共有でホームディレクトリの自動作成機能を有効にするかどうかを設定します。</p> <p>true ホームディレクトリの自動作成機能を有効にする場合に指定します。</p> <p>false<sup>*1</sup> ホームディレクトリの自動作成機能を無効にする場合に指定します。</p>	任意
	clientAccessPolicy	<p>CIFS クライアントからのアクセスの処理方法を設定します。</p> <p>parallel アクセスをパラレルに処理する場合に指定します。</p> <p>serial アクセスをシリアルに処理する場合に指定します。</p> <p>default<sup>*1</sup> GUI の [CIFS Service Management] ページ (Setting Type : Performance) で設定した CIFS サービス構成定義に従う場合に指定します。</p>	任意



プロパティ		説明	指定時の条件
	vssUse	<p>Volume Shadow Copy Service を使用するかどうかを指定します。ただし、最新差分スナップショット公開用ファイルシステムの CIFS 共有の場合は、このプロパティの設定にかかわらず、do_not_use が設定されます。</p> <p>default*1</p> <p>GUI の [CIFS Service Management] ページ (Setting Type : Basic) で設定した CIFS サービス構成定義に従う場合に指定します。</p> <p>do_not_use</p> <p>Volume Shadow Copy Service を使用しない場合に指定します。</p>	任意
	cifsClientRoCacheOptimize	<p>複数の CIFS クライアントによるファイルへのアクセス競合が発生した場合に、読み取り専用のクライアントキャッシュを使用するかどうかを設定します。なお、ファイル共有に NFS プロトコルでもアクセスする場合は、更新が反映されないおそれがあるため、読み取り専用のクライアントキャッシュを使用しないことを推奨します。</p> <p>use</p> <p>読み取り専用のクライアントキャッシュを使用する場合に指定します。</p> <p>do_not_use</p> <p>読み取り専用のクライアントキャッシュを使用しない場合に指定します。</p> <p>default*1</p> <p>GUI の [CIFS Service Management] ページ (Setting Type : Performance) で設定した CIFS サービス構成定義に従う場合に指定します。</p>	任意

プロパティ		説明	指定時の条件
	accessBasedEnumeration	<p>アクセススペースの列挙を有効にするかどうかを設定します。</p> <p>use アクセススペースの列挙を有効にする場合に指定します。</p> <p>do_not_use アクセススペースの列挙を無効にする場合に指定します。</p> <p>default<sup>*1</sup> GUI の [CIFS Service Management] ページ (Setting Type : Security) で設定した CIFS サービス構成定義に従う場合に指定します。</p>	任意
	writeDisallowedUsers*3	CIFS 共有への書き込みを拒否するユーザーを設定します。複数のユーザーが存在する場合は、コンマ (,) で区切って指定します。省略した場合、値は設定されません。	任意
	writeDisallowedGroups*3	CIFS 共有への書き込みを拒否するグループを設定します。複数のグループが存在する場合は、コンマ (,) で区切って指定します。省略した場合、値は設定されません。	任意
	writeAllowedUsers*3	CIFS 共有への書き込みを許可するユーザーを設定します。複数のユーザーが存在する場合は、コンマ (,) で区切って指定します。省略した場合、値は設定されません。	任意
	writeAllowedGroups*3	CIFS 共有への書き込みを許可するグループを設定します。複数のグループが存在する場合は、コンマ (,) で区切って指定します。省略した場合、値は設定されません。	任意
	DirectorySetup	<p>共有の作成と同時に、共有ディレクトリを作成する場合に指定します。</p> <p>このプロパティで設定した内容を確認するには、作成した共有ディレクトリにクライアントからアクセスできるかどうかを確認してください。</p>	任意

プロパティ			説明	指定時の条件
	--	userName	<p>作成するディレクトリのオーナーのユーザー名またはユーザー ID を設定します。</p> <p>ユーザー ID で設定する場合は、0 ～ 2147483147 の範囲で指定してください。ただし、「0」以外の ID については、先頭に 0 を指定できません。</p> <p>Windows ドメインのビルトインユーザーは指定できません。</p> <p>省略した場合、userName プロパティおよび groupName プロパティには「root」が設定され、isStickyBit プロパティ、ownerPermission プロパティ、groupPermission プロパティ、および otherPermission プロパティのアクセス権は設定されません。</p>	DirectorySetup プロパティを設定する場合に指定できます。
		groupName	<p>ディレクトリを所有するグループの名称または ID を設定します。</p> <p>グループ ID で設定する場合は、0 ～ 2147483147 の範囲で指定してください。ただし、「0」以外の ID については、先頭に 0 を指定できません。Windows ドメインのビルトイングループは指定できません。</p> <p>省略した場合、userName プロパティおよび groupName プロパティには「root」が設定され、isStickyBit プロパティ、ownerPermission プロパティ、groupPermission プロパティ、および otherPermission プロパティのアクセス権は設定されません。</p>	DirectorySetup プロパティを設定する場合に指定できます。
		isStickyBit	<p>スティッキービットを使用するかどうかを設定します。</p> <p>このプロパティの指定は、userName プロパティおよび groupName プロパティを設定する場合にだけ有効です。</p> <p>true</p> <p>スティッキービットを使用する場合に指定します。</p> <p>false<sup>*1</sup></p> <p>スティッキービットを使用しない場合に指定します。</p>	DirectorySetup プロパティを設定する場合に指定できます。

プロパティ		説明	指定時の条件
	ownerPermission	共有ディレクトリへのオーナーのアクセス権を設定します。 このプロパティの指定は、 userName プロパティおよび groupName プロパティを設定する場合にだけ有効です。 省略した場合、 ownerPermission プロパティにアクセス権は設定されませんが、isStickyBit プロパティ、groupPermission プロパティ、および otherPermission プロパティをすべて省略すると、ownerPermission プロパティには rw が設定されます。 設定するアクセス権については、fileAccessPermissionOfOwner プロパティを参照してください。	DirectorySetup プロパティを設定する場合に指定できます。
	groupPermission	共有ディレクトリへのグループのアクセス権を設定します。 このプロパティの指定は、 userName プロパティおよび groupName プロパティを設定する場合にだけ有効です。 省略した場合、 groupPermission プロパティにアクセス権は設定されませんが、isStickyBit プロパティ、ownerPermission プロパティ、および otherPermission プロパティをすべて省略すると、groupPermission プロパティには ro が設定されます。 設定するアクセス権については、fileAccessPermissionOfOwner プロパティを参照してください。	DirectorySetup プロパティを設定する場合に指定できます。

プロパティ			説明	指定時の条件
		otherPermission	<p>共有ディレクトリへのその他のアクセス権を設定します。</p> <p>このプロパティの指定は、userName プロパティおよび groupName プロパティを設定する場合にだけ有効です。</p> <p>省略した場合、otherPermission プロパティにアクセス権は設定されませんが、isStickyBit プロパティ、ownerPermission プロパティ、および groupPermission プロパティをすべて省略すると、otherPermission プロパティには ro が設定されます。</p> <p>設定するアクセス権については、fileAccessPermissionOfOwner プロパティを参照してください。</p>	DirectorySetup プロパティを設定する場合に指定できます。

プロパティ			説明	指定時の条件
	DirectoryACLSettings		共有ディレクトリに ACL を設定する場合に指定します。 このプロパティで設定した内容を確認するには、作成した共有ディレクトリにクライアントからアクセスできるかどうかを確認してください。	任意
	--	DirectoryACLSetting	--	DirectoryACLSettings プロパティを設定する場合に必須です。
	--	userGroupAccess	ACE にユーザーまたはグループのアクセス権を設定します。  allow 指定したユーザーまたはグループに許可のアクセス権を与える場合に指定します。ユーザーまたはグループの ACE にアクセス権が追加され、すでに ACE が設定されている場合は、アクセス権が統合されます。同一ユーザーに対して複数の ACE が設定されている場合は、一つの ACE に統合されます。  replace 指定したユーザーまたはグループに許可のアクセス権を与える場合に指定します。同一ユーザーまたはグループの ACE は破棄され、指定した ACE に置き換えられます。  deny 指定したユーザーまたはグループに拒否のアクセス権を与える場合に指定します。同一ユーザーまたはグループの ACE は破棄され、指定した ACE に置き換えられます。	DirectoryACLSettings プロパティを設定する場合に必須です。
		name	ACE のユーザー名またはグループ名を設定します。	DirectoryACLSettings プロパティを設定する場合に必須です。
		accountType	ACE のアカウント種別を設定します。  u ユーザーにアクセス権を与える場合に指定します。  g グループにアクセス権を与える場合に指定します。	DirectoryACLSetting プロパティが設定されている場合は必須

プロパティ			説明	指定時の条件
		applicationDestination	<p>ACE が適用される範囲を設定します。</p> <ul style="list-style-type: none"> <li>o ディレクトリだけに適用する場合に指定します。</li> <li>a ディレクトリ、サブディレクトリおよびファイルだけに適用する場合に指定します。</li> <li>f ディレクトリとサブディレクトリに適用する場合に指定します。</li> <li>e ディレクトリとファイルに適用する場合に指定します。</li> <li>s サブディレクトリとファイルだけに適用する場合に指定します。</li> <li>u サブディレクトリだけに適用する場合に指定します。</li> <li>i ファイルだけに適用する場合に指定します。</li> </ul>	DirectoryACLSetting プロパティが設定されている場合は必須
		inheritanceRange	<p>ACE の継承範囲を設定します。</p> <ul style="list-style-type: none"> <li>o 直下にあるディレクトリやファイルにだけ継承する場合に指定します。</li> <li>a 下位にあるすべてのディレクトリおよびファイルに継承する場合に指定します。</li> </ul>	DirectoryACLSettings プロパティを設定する場合に必須です。

プロパティ			説明	指定時の条件
		accessPermission	<p>アクセス権を設定します。複数のアクセス権を設定する場合は、コロン二つ (:) で区切って指定してください。</p> <p>f 「フルコントロール」を許可する場合に指定します。</p> <p>s 「フォルダのスキャン/ファイルの実行」を許可する場合に指定します。</p> <p>r 「フォルダの一覧/データの読み取り」を許可する場合に指定します。</p> <p>a 「属性の読み取り」を許可する場合に指定します。</p> <p>e 「拡張属性の読み取り」を許可する場合に指定します。</p> <p>w 「ファイルの作成/データの書き込み」を許可する場合に指定します。</p> <p>p 「フォルダの作成/データの追加」を許可する場合に指定します。</p> <p>t 「属性の書き込み」を許可する場合に指定します。</p> <p>x 「拡張属性の書き込み」を許可する場合に指定します。</p> <p>l 「サブフォルダとファイルの削除」を許可する場合に指定します。</p> <p>d 「削除」を許可する場合に指定します。</p> <p>c 「アクセス許可の読み取り」を許可する場合に指定します。</p> <p>h 「アクセス許可の変更」を許可する場合に指定します。</p> <p>g 「所有権の取得」を許可する場合に指定します。</p>	DirectoryACLSettings プロパティを設定する場合に必須です。

注 \*1 プロパティを省略した場合のデフォルト値です。



注 \*2 Advanced ACL タイプのファイルシステムの場合、次に示すプロパティをすべて空文字列で指定すると、cifscreate コマンドで --owner-only オプションを指定したときと同じ設定となります。

- fileAccessPermissionOfOwner プロパティ
- fileAccessPermissionOfGroup プロパティ
- fileAccessPermissionOfOther プロパティ
- directoryAccessPermissionOfOwner プロパティ
- directoryAccessPermissionOfGroup プロパティ
- directoryAccessPermissionOfOther プロパティ

注 \*3 ユーザーまたはグループにアクセス権を設定する場合、ユーザー数を  $u$ 、グループ数を  $g$ 、ユーザー名の総文字数を  $n$ 、グループ名の総文字数を  $m$  とすると、次の式を満たす必要があります。

$$u+2g+n+m \leq 1024$$

CIFSShare リソースに PUT メソッドをリクエストして、CIFS 共有を追加する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/CIFSShares

### リクエスト XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CIFSShare>
  <exportPoint>/mnt/fs/cifs</exportPoint>
  <cifsShareName>cifsshare</cifsShareName>
  <isClassicAcl>>false</isClassicAcl>
  <isAclEnabled>>true</isAclEnabled>
  <restrictionTargetHosts>10.213.77.231</restrictionTargetHosts>
  <isAccessRestrictionAllowed>>true</isAccessRestrictionAllowed>
  <commentForShare>cifscomment</commentForShare>
  <isReadOnly>>false</isReadOnly>
  <isBrowseEnabled>>true</isBrowseEnabled>
  <guestAccessPermission>default</guestAccessPermission>
  <diskSynchronizationPolicy>default</diskSynchronizationPolicy>
  <cifsClientCacheSetting>default</cifsClientCacheSetting>
  <isHomeDirectoryEnabled>>true</isHomeDirectoryEnabled>
  <clientAccessPolicy>default</clientAccessPolicy>
  <vssUse>default</vssUse>
  <cifsClientRoCacheOptimize>default</cifsClientRoCacheOptimize>
  <accessBasedEnumeration>default</accessBasedEnumeration>
  <writeDisallowedUsers>user3,user4</writeDisallowedUsers>
  <writeDisallowedGroups>group3,group4</writeDisallowedGroups>
  <writeAllowedUsers>user1,user2</writeAllowedUsers>
  <writeAllowedGroups>group1,group2</writeAllowedGroups>
  <DirectorySetup>
    <userName>22222</userName>
    <groupName>11111</groupName>
    <isStickyBit>>true</isStickyBit>
    <ownerPermission>rw</ownerPermission>
    <groupPermission>rw</groupPermission>
    <otherPermission>rw</otherPermission>
  </DirectorySetup>
  <DirectoryACLSettings>
    <DirectoryACLSetting>
      <userGroupAccess>allow</userGroupAccess>
      <name>Everyone</name>
      <accountType>g</accountType>
      <applicationDestination>a</applicationDestination>
      <inheritanceRange>a</inheritanceRange>
      <accessPermission>r</accessPermission>
    </DirectoryACLSetting>
  </DirectoryACLSettings>
</CIFSShare>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-15: CIFSShare リソースに PUT メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3: 」を参照してください。

### 3.3.3 CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソース

この項では、CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースの使い方について説明します。

#### 3.3.3.1 GET メソッドで CIFS 共有の情報を取得する

クエリパラメーター SHARE-DIRECTORY-NAME を使用して、Base64 でエンコードされた共有ディレクトリのパス名を必ず指定してください。

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

表 3-16: CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ		出力される個数	サポート構成	verbose が false の場合の表示
CIFSShare		1	クラスタ Virtual Server	○
--	exportPoint	1	クラスタ Virtual Server	○
	cifsShareName	1	クラスタ Virtual Server	○
	isAclEnabled	1	クラスタ Virtual Server	○
	restrictionTargetHosts	1	クラスタ Virtual Server	○
	isAccessRestrictionAllowed	1	クラスタ Virtual Server	○
	commentForShare	1	クラスタ Virtual Server	○
	isReadOnly	1	クラスタ Virtual Server	○
	isBrowseEnabled	1	クラスタ Virtual Server	○
	fileAccessPermissionOfOwner	1	クラスタ Virtual Server	○
	fileAccessPermissionOfGroup	1	クラスタ Virtual Server	○
	fileAccessPermissionOfOther	1	クラスタ Virtual Server	○
	directoryAccessPermissionOfOwner	1	クラスタ Virtual Server	○
	directoryAccessPermissionOfGroup	1	クラスタ Virtual Server	○

プロパティ		出力される個 数	サポート構成	verbose が false の 場合の表示
	directoryAccessPermissionOfOther	1	クラスタ Virtual Server	○
	guestAccessPermission	1	クラスタ Virtual Server	○
	diskSynchronizationPolicy	1	クラスタ Virtual Server	○
	cifsClientCacheSetting	1	クラスタ Virtual Server	○
	fileTimestampChangeableUsers	1	クラスタ Virtual Server	○
	isHomeDirectoryEnabled	1	クラスタ Virtual Server	○
	aclType	1	クラスタ Virtual Server	×
	clientAccessPolicy	1	クラスタ Virtual Server	○
	vssUse	1	クラスタ Virtual Server	○
	cifsClientRoCacheOptimize	1	クラスタ Virtual Server	○
	accessBasedEnumeration	1	クラスタ Virtual Server	○
	writeDisallowedUsers	1	クラスタ Virtual Server	○
	writeDisallowedGroups	1	クラスタ Virtual Server	○
	writeAllowedUsers	1	クラスタ Virtual Server	○
	writeAllowedGroups	1	クラスタ Virtual Server	○

(凡例) ○ : 表示されます。× : 表示されません。

CIFSShares/<Base64 でエンコードされた CIFS 共有名>リソースに GET メソッドをリクエストしたときに表示されるプロパティについては、「[表 3-11 : CIFSShare リソースに GET メソッドをリクエストしたときのプロパティ](#)」を参照してください。

CIFSShares/<Base64 でエンコードされた CIFS 共有名>リソースに GET メソッドをリクエストして、指定した CIFS 共有の情報を取得する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/CIFSShares/<Base64でエンコードされたCIFS共有名>

### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CIFSShare>
  <exportPoint>/mnt/aaa/sh</exportPoint>
  <cifsShareName>sh</cifsShareName>
  <isAclEnabled>true</isAclEnabled>
  <restrictionTargetHosts>0.0.0.0/0</restrictionTargetHosts>
  <isAccessRestrictionAllowed>false</isAccessRestrictionAllowed>
  <commentForShare></commentForShare>
  <isReadOnly>false</isReadOnly>
  <isBrowseEnabled>true</isBrowseEnabled>
  <fileAccessPermissionOfOwner></fileAccessPermissionOfOwner>
</CIFSShare>
```

```

<fileAccessPermissionOfGroup></fileAccessPermissionOfGroup>
<fileAccessPermissionOfOther></fileAccessPermissionOfOther>
<directoryAccessPermissionOfOwner></directoryAccessPermissionOfOwner>
<directoryAccessPermissionOfGroup></directoryAccessPermissionOfGroup>
<directoryAccessPermissionOfOther></directoryAccessPermissionOfOther>
<guestAccessPermission>default</guestAccessPermission>
<diskSynchronizationPolicy>default</diskSynchronizationPolicy>
<cifsClientCacheSetting>default</cifsClientCacheSetting>
<fileTimestampChangeableUsers></fileTimestampChangeableUsers>
<isHomeDirectoryEnabled>false</isHomeDirectoryEnabled>
<clientAccessPolicy>default</clientAccessPolicy>
<vssUse>default</vssUse>
<cifsClientRoCacheOptimize>default</cifsClientRoCacheOptimize>
<accessBasedEnumeration>default</accessBasedEnumeration>
<writeDisallowedUsers></writeDisallowedUsers>
<writeDisallowedGroups></writeDisallowedGroups>
<writeAllowedUsers></writeAllowedUsers>
<writeAllowedGroups></writeAllowedGroups>
</CIFSShare>

```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-17 : CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに GET メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.3.3.2 POST メソッドで CIFS 共有の情報を変更する

クエリパラメーター SHARE-DIRECTORY-NAME を使用して、Base64 でエンコードされた共有ディレクトリのパス名を必ず指定してください。

CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに POST メソッドをリクエストするときの XML の構造を次の表に示します。

**表 3-18 : CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに POST メソッドをリクエストするときの XML の構造**

プロパティ		データ型	指定できる個数	サポート構成
CIFSShare		--	1	クラスタ Virtual Server
--	cifsShareName	String	0, 1	クラスタ Virtual Server
	isClassicAcl	Boolean	1	クラスタ Virtual Server
	isAclEnabled	Boolean	0, 1	クラスタ Virtual Server
	restrictionTargetHosts	String	0, 1	クラスタ Virtual Server
	isAccessRestrictionAllowed	Boolean	0, 1	クラスタ Virtual Server
	commentForShare	String	0, 1	クラスタ Virtual Server

プロパティ		データ型	指定できる個数	サポート構成
	isReadOnly	Boolean	0, 1	クラスタ Virtual Server
	isBrowseEnabled	Boolean	0, 1	クラスタ Virtual Server
	fileAccessPermissionOfOwner	String	0, 1	クラスタ Virtual Server
	fileAccessPermissionOfGroup	String	0, 1	クラスタ Virtual Server
	fileAccessPermissionOfOther	String	0, 1	クラスタ Virtual Server
	directoryAccessPermissionOfOwner	String	0, 1	クラスタ Virtual Server
	directoryAccessPermissionOfGroup	String	0, 1	クラスタ Virtual Server
	directoryAccessPermissionOfOther	String	0, 1	クラスタ Virtual Server
	guestAccessPermission	String	0, 1	クラスタ Virtual Server
	diskSynchronizationPolicy	String	0, 1	クラスタ Virtual Server
	cifsClientCacheSetting	String	0, 1	クラスタ Virtual Server
	fileTimestampChangeableUsers	String	0, 1	クラスタ Virtual Server
	isHomeDirectoryEnabled	Boolean	0, 1	クラスタ Virtual Server
	clientAccessPolicy	String	0, 1	クラスタ Virtual Server
	vssUse	String	0, 1	クラスタ Virtual Server
	cifsClientRoCacheOptimize	String	0, 1	クラスタ Virtual Server
	accessBasedEnumeration	String	0, 1	クラスタ Virtual Server
	writeDisallowedUsers	String	0, 1	クラスタ Virtual Server
	writeDisallowedGroups	String	0, 1	クラスタ Virtual Server
	writeAllowedUsers	String	0, 1	クラスタ Virtual Server
	writeAllowedGroups	String	0, 1	クラスタ Virtual Server

CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに POST メソッドをリクエストするときに指定するプロパティについて次の表に示します。

**表 3-19 : CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに POST メソッドをリクエストするときのプロパティ**

プロパティ	説明	指定時の条件
CIFSShare	--	--

プロパティ		説明	指定時の条件
--	cifsShareName	CIFS 共有名を設定します。 設定できる文字列の長さは、80 文字以内です。	任意
	isClassicAcl	Advanced ACL または Classic ACL のどちらで操作するかを設定します。操作対象の共有ディレクトリが属するファイルシステムの ACL タイプに合わせて設定してください。 true Classic ACL タイプのファイルシステムの場合に指定します。 false Advanced ACL タイプのファイルシステムの場合に指定します。	必須
	isAclEnabled	CIFS クライアントに ACL の参照・設定を許可するかどうかを指定します。 true から false には変更できません。 true CIFS クライアントに ACL の参照・設定を許可する場合に指定します。 false CIFS クライアントに ACL の参照・設定を許可しない場合に指定します。	任意

プロパティ	説明	指定時の条件
restrictionTargetHosts	<p>CIFS 共有へのアクセスを制限するホストを設定します。複数指定する場合はコンマ (,) で区切ってください。</p> <p>ホストをネットワークアドレスで指定する場合は、IP アドレスを指定します (例 : 10.203.15.0)。</p> <p>ネットマスクに従ってネットワークの範囲を指定する場合は、&lt;ネットワークアドレス&gt; / &lt;ネットマスク&gt; の形式で指定します (例 : 10.203.15.0/255.255.255.0)。</p> <p>ホスト名を指定する場合は、GUI の [Edit System File] ページで /etc/hosts ファイルを編集して、指定したすべてのホスト名と IP アドレスを追加してください。</p> <p>省略した場合、すべてのホストまたはネットワークが対象になります。</p> <p>GUI の [CIFS Service Management] ページ (Setting Type : Security) でアクセスを制限するホストまたはネットワークが設定されている場合は、すべてのファイル共有にその設定が適用されます。</p>	isAccessRestrictionAllowed プロパティを設定する場合に必須です。
isAccessRestrictionAllowed	<p>restrictionTargetHosts プロパティで指定したホストに対して CIFS 共有へのアクセスを制限するかどうかを設定します。</p> <p>true アクセスを制限する場合に指定します。</p> <p>false アクセスを制限しない場合に指定します。</p> <p>GUI の [CIFS Service Management] ページ (Setting Type : Security) でアクセスを制限するホストまたはネットワークが設定されている場合は、すべてのファイル共有にその設定が適用されます。</p>	restrictionTargetHosts プロパティを設定する場合に必須です。
commentForShare	<p>CIFS 共有のコメントを設定します。</p> <p>設定できる文字は、英数字、感嘆符 (!), 番号記号 (#), ドル記号 (\$), アンパサンド (&amp;), アポストロフィ ('), 始め丸括弧 ((), 終わり丸括弧 ()), アステリクス (*), 正符号 (+), コンマ (,), ハイフン (-), ピリオド (.), 斜線 (/), コロン (:), 始め山括弧 (&lt;), 終わり山括弧 (&gt;), 疑問符 (?), 単価記号 (@), 始め角括弧 ([), 円記号 (¥), 終わり角括弧 (]), アクサンシルコンフレックス (^), アンダーライン (_), アクサングラーブ (`), 始め波括弧 ({), 縦線 ( ), 終わり波括弧 (}), および波ダッシュ (~) です。スペースも指定できますが、文字列の先頭および末尾には指定できません。また、円記号 (¥) は文字列の末尾には指定できません。このほか、マルチバイト文字も指定できます。設定できる文字列の長さは、256 文字以内です。</p>	任意

プロパティ		説明	指定時の条件
	isReadOnly	CIFS 共有を読み取り専用にするかどうかを設定します。 true 読み取り専用にする場合に設定します。 false 読み取りと書き込みを許可する場合に設定します。	任意
	isBrowseEnabled	クライアント環境で CIFS 共有名を一覧に表示するかどうかを設定します。 true 表示する場合に指定します。 false 表示しない場合に指定します。	任意
	fileAccessPermissionOfOwner*	書き込みが許可された CIFS 共有にファイルを作成するときのオーナーのアクセス権を指定します。 rw 読み取りおよび書き込みを許可する場合に指定します。 ro 読み取りだけを許可する場合に指定します。 none 読み取りおよび書き込みを許可しない場合に指定します。	fileAccessPermissionOfGroup プロパティおよび fileAccessPermissionOfOther プロパティが設定されている場合は必須です。
	fileAccessPermissionOfGroup*	書き込みが許可された CIFS 共有にファイルを作成するときのグループのアクセス権を指定します。 設定するアクセス権については、fileAccessPermissionOfOwner プロパティを参照してください。	fileAccessPermissionOfOwner プロパティおよび fileAccessPermissionOfOther プロパティが設定されている場合は必須です。



プロパティ	説明	指定時の条件
fileAccessPermissionOfOther*	書き込みが許可された CIFS 共有にファイルを作成するときのその他のアクセス権を指定します。 設定するアクセス権については、fileAccessPermissionOfOwner プロパティを参照してください。	fileAccessPermissionOfOwner プロパティおよび fileAccessPermissionOfGroup プロパティが設定されている場合は必須です。
directoryAccessPermissionOfOwner*	書き込みが許可された CIFS 共有にディレクトリを作成するときのオーナーのアクセス権を指定します。 設定するアクセス権については、fileAccessPermissionOfOwner プロパティを参照してください。	directoryAccessPermissionOfGroup プロパティおよび directoryAccessPermissionOfOther プロパティが設定されている場合は必須です。
directoryAccessPermissionOfGroup*	書き込みが許可された CIFS 共有にディレクトリを作成するときのグループのアクセス権を指定します。 設定するアクセス権については、fileAccessPermissionOfOwner プロパティを参照してください。	directoryAccessPermissionOfOwner プロパティおよび directoryAccessPermissionOfOther プロパティが設定されている場合は必須です。

プロパティ	説明	指定時の条件
directoryAccessPermissionOfOther*	書き込みが許可された CIFS 共有にディレクトリを作成するときのその他のアクセス権を指定します。 設定するアクセス権については、fileAccessPermissionOfOwner プロパティを参照してください。	directoryAccessPermissionOfOwner プロパティおよび directoryAccessPermissionOfGroup プロパティが設定されている場合は必須です。
guestAccessPermission	ゲストアカウントのアクセスを許可するかどうかを設定します。 allow CIFS 共有にゲストアカウントでのアクセスを許可する場合に指定します。 disallow CIFS 共有にゲストアカウントでのアクセスを許可しない場合に指定します。 default GUI の [CIFS Service Management] ページ (Setting Type : Security) で設定した CIFS サービス構成定義に従う場合に指定します。	任意
diskSynchronizationPolicy	クライアントからの書き込みまたはクローズ要求に対する動作を設定します。 write_and_close クライアントから CIFS 共有への書き込みおよびクローズ要求に同期して書き込む場合に指定します。 close クライアントから CIFS 共有へのクローズ要求に同期して書き込む場合に指定します。 none 一定周期で書き込む場合に指定します。 default GUI の [CIFS Service Management] ページ (Setting Type : Performance) で設定した CIFS サービス構成定義に従う場合に指定します。	任意

プロパティ	説明	指定時の条件
cifsClientCacheSetting	<p>CIFS 共有内のファイルの更新データをクライアントにキャッシュするかどうかを設定します。</p> <p>use クライアントにキャッシュする場合に指定します。</p> <p>do_not_use クライアントにキャッシュしない場合に指定します。</p> <p>default GUI の [CIFS Service Management] ページ (Setting Type : Performance) で設定した CIFS サービス構成定義に従う場合に指定します。</p>	任意
fileTimestampChangeableUsers	<p>CIFS 共有内のファイルのタイムスタンプを更新できるユーザーを設定します。</p> <p>write_permitted_users ファイルの書き込みを許可されたすべてのユーザーに、タイムスタンプの更新を許可する場合に指定します。</p> <p>owner_only ファイルのオーナーだけにタイムスタンプの更新を許可する場合に指定します。</p> <p>default GUI の [CIFS Service Management] ページ (Setting Type : Security) で設定した CIFS サービス構成定義に従う場合に指定します。</p>	isClassicAc1 プロパティが true の場合に指定できます。
isHomeDirectoryEnabled	<p>CIFS 共有でホームディレクトリの自動作成機能を有効にするかどうかを設定します。</p> <p>true ホームディレクトリの自動作成機能を有効にする場合に指定します。</p> <p>false ホームディレクトリの自動作成機能を無効にする場合に指定します。</p>	任意
clientAccessPolicy	<p>CIFS クライアントからのアクセスの処理方法を設定します。</p> <p>parallel アクセスをバラレルに処理する場合に指定します。</p> <p>serial アクセスをシリアルに処理する場合に指定します。</p> <p>default GUI の [CIFS Service Management] ページ (Setting Type : Performance) で設定した CIFS サービス構成定義に従う場合に指定します。</p>	任意

プロパティ	説明	指定時の条件
vssUse	<p>Volume Shadow Copy Service を使用するかどうかを指定します。ただし、最新差分スナップショット公開用ファイルシステムの CIFS 共有の場合は、このプロパティの設定に関わらず、do_not_use が設定されます。</p> <p>default</p> <p>GUI の [CIFS Service Management] ページ (Setting Type : Basic) で設定した CIFS サービス構成定義に従う場合に指定します。</p> <p>do_not_use</p> <p>Volume Shadow Copy Service を使用しない場合に指定します。</p>	任意
cifsClientRoCacheOptimize	<p>複数の CIFS クライアントによるファイルへのアクセス競合が発生した場合に、読み取り専用のクライアントキャッシュを使用するかどうかを設定します。なお、ファイル共有に NFS プロトコルでもアクセスする場合は、更新が反映されないおそれがあるため、読み取り専用のクライアントキャッシュを使用しないことを推奨します。</p> <p>use</p> <p>読み取り専用のクライアントキャッシュを使用する場合に指定します。</p> <p>do_not_use</p> <p>読み取り専用のクライアントキャッシュを使用しない場合に指定します。</p> <p>default</p> <p>GUI の [CIFS Service Management] ページ (Setting Type : Performance) で設定した CIFS サービス構成定義に従う場合に指定します。</p>	任意
accessBasedEnumeration	<p>アクセスベースの列挙を有効にするかどうかを設定します。</p> <p>use</p> <p>アクセスベースの列挙を有効にする場合に指定します。</p> <p>do_not_use</p> <p>アクセスベースの列挙を無効にする場合に指定します。</p> <p>default</p> <p>GUI の [CIFS Service Management] ページ (Setting Type : Security) で設定した CIFS サービス構成定義に従う場合に指定します。</p>	任意

プロパティ		説明	指定時の条件
	writeDisallowedUsers	CIFS 共有への書き込みを拒否するユーザーを設定します。 複数のユーザーが存在する場合は、コンマ (,) で区切って指定します。 プロパティの設定を省略した場合、または空文字列を指定した場合は、設定されていた情報が削除されます。ただし、writeDisallowedGroups プロパティの設定も省略した場合、または空文字列を指定した場合は、設定されていた情報は削除されません。	任意
	writeDisallowedGroups	CIFS 共有への書き込みを拒否するグループを設定します。 複数のグループが存在する場合は、コンマ (,) で区切って指定します。 プロパティの設定を省略した場合、または空文字列を指定した場合は、設定されていた情報が削除されます。ただし、writeDisallowedUsers プロパティの設定も省略した場合、または空文字列を指定した場合は、設定されていた情報は削除されません。	任意
	writeAllowedUsers	CIFS 共有への書き込みを許可するユーザーを設定します。 複数のユーザーが存在する場合は、コンマ (,) で区切って指定します。 プロパティの設定を省略した場合、または空文字列を指定した場合は、設定されていた情報が削除されます。ただし、writeAllowedGroups プロパティの設定も省略した場合、または空文字列を指定した場合は、設定されていた情報は削除されません。	任意
	writeAllowedGroups	CIFS 共有への書き込みを許可するグループを設定します。 複数のグループが存在する場合は、コンマ (,) で区切って指定します。 プロパティの設定を省略した場合、または空文字列を指定した場合は、設定されていた情報が削除されます。ただし、writeAllowedUsers プロパティの設定も省略した場合、または空文字列を指定した場合は、設定されていた情報は削除されません。	任意

注 \* Advanced ACL タイプのファイルシステムの場合、次に示すプロパティをすべて空文字列で指定すると、cifsedit コマンドで --owner-only オプションを指定したときと同じ設定となります。

- fileAccessPermissionOfOwner プロパティ
- fileAccessPermissionOfGroup プロパティ
- fileAccessPermissionOfOther プロパティ
- directoryAccessPermissionOfOwner プロパティ
- directoryAccessPermissionOfGroup プロパティ
- directoryAccessPermissionOfOther プロパティ

CIFSShares/< Base64 でエンコードされた CIFS 共有名>リソースに POST メソッドをリクエストして、CIFS 共有の情報を変更する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/CIFSShares/<Base64でエンコードされたCIFS共有名>

### リクエスト XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CIFSShare>
  <cifsShareName>cifssharechange</cifsShareName>
  <isClassicAcl>false</isClassicAcl>
  <isAclEnabled>true</isAclEnabled>
  <restrictionTargetHosts>10.213.77.88</restrictionTargetHosts>
  <isAccessRestrictionAllowed>true</isAccessRestrictionAllowed>
  <commentForShare>cifscomment</commentForShare>
  <isReadOnly>false</isReadOnly>
  <isBrowseEnabled>true</isBrowseEnabled>
  <guestAccessPermission>default</guestAccessPermission>
  <diskSynchronizationPolicy>default</diskSynchronizationPolicy>
  <cifsClientCacheSetting>default</cifsClientCacheSetting>
  <isHomeDirectoryEnabled>true</isHomeDirectoryEnabled>
  <clientAccessPolicy>default</clientAccessPolicy>
  <vssUse>default</vssUse>
  <cifsClientRoCacheOptimize>default</cifsClientRoCacheOptimize>
  <accessBasedEnumeration>default</accessBasedEnumeration>
  <writeDisallowedUsers>user1,user2</writeDisallowedUsers>
  <writeDisallowedGroups>group1,group2</writeDisallowedGroups>
  <writeAllowedUsers>user3,user4</writeAllowedUsers>
  <writeAllowedGroups>group3,group4</writeAllowedGroups>
</CIFSShare>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-20 : CIFSShares/ < Base64 でエンコードされた CIFS 共有名>リソースに POST メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.3.3.3

### DELETE メソッドで指定した CIFS 共有を削除する

クエリパラメーター SHARE-DIRECTORY-NAME を使用して、Base64 でエンコードされた共有ディレクトリのパス名を必ず指定してください。

CIFS 共有を削除しても、共有ディレクトリは削除されません。

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-21 : CIFSShares/ < Base64 でエンコードされた CIFS 共有名>リソースに DELETE メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.3.3.4

### HEAD メソッドで指定した CIFS 共有が存在するかどうか確認する

クエリパラメーター SHARE-DIRECTORY-NAME を使用して、Base64 でエンコードされた共有ディレクトリのパス名を必ず指定してください。

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-22 : CIFSShares/ < Base64 でエンコードされた CIFS 共有名 > リソースに HEAD メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	指定した CIFS 共有は利用できます。
その他	異常終了しました。「表 1-3 : 」を参照してください。

## 3.4 FCPath

### 3.4.1 FCPath リソースの概要

FCPath リソースを使用して、FC パスの情報を取得できます。

FCPath リソースに使用できる HTTP メソッドを次の表に示します。

表 3-23 : FCPath リソースに使用できる HTTP メソッド

リソースの URI	HTTP メソッド	サポート構成	説明
/FCPaths	GET	クラスタ	ログインしたノードに設定されている全 FC パスの情報を取得します。

### 3.4.2 FCPath リソース

この項では、FCPath リソースの使い方について説明します。

#### 3.4.2.1 GET メソッドで指定した FC パスの情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

表 3-24 : FCPath リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ		出力される個数	サポート構成	verbose が false の場合の表示
FCPaths		1	クラスタ	○
--	FCPath	0, 1 ~ n	クラスタ	○
	-- path	1	クラスタ	○
	target	1	クラスタ	○
	hostPort	1	クラスタ	○
	hostPortWWN	1	クラスタ	○
	arrayPort	1	クラスタ	○
	arrayPortWWN	1	クラスタ	○
	model	1	クラスタ	○
	serial	1	クラスタ	○
	status	1	クラスタ	○

(凡例) n : 2 以上の整数を示します。○ : 表示されます。

FCPath リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

表 3-25：FCPath リソースに GET メソッドをリクエストしたときのプロパティ

プロパティ			説明
FCPaths			--
--	FCPath		--
	--	path	FC パスが次の形式で表示されます。 path <固定 GID> - <ホストポート番号> - <ストレージポート番号> ターゲットを特定できない場合は、空文字列が表示されます。
	--	target	ターゲットが 7 バイトで表示されます。 ターゲットを特定できない場合は、空文字列が表示されます。
	--	hostPort	ノード側の FC ポート（ホストポート）の名称が 6 バイトで表示されます。 FC ポートを特定できない場合は、空文字列が表示されます。
	--	hostPortWWN	ノード側の FC ポートの WWN が 16 バイトで表示されます。 FC ポートを特定できない場合は、「-----」が表示されます。



プロパティ			説明
		arrayPort	ストレージシステム側の FC ポート（ストレージポート）の名称が 4 バイト以内で表示されます。
		arrayPortWWN	ストレージシステム側の FC ポートの WWN が表示されます。 FC ポートを特定できない場合は、「-----」が表示されます。
		model	LU が存在するストレージシステムのモデルが表示されます。 次の場合は、空文字列が表示されます。 <ul style="list-style-type: none"> <li>・ HVFP で LU を運用していないストレージシステムである</li> <li>・ ノード上の OS の起動時に FC パスに障害が発生しているなどの理由でストレージシステムの情報が取得されていない</li> </ul>
		serial	ストレージシステムのシリアル番号が 16 バイト以内で表示されます。 特定できない場合は空文字列が表示されます。
		status	FC パスの状態が表示されます。 Online 正常に稼働している場合に表示されます。 デバイスファイルの障害を検知した場合は、Online (LU Error) と表示されます。 Offline システム管理者の操作によって FC パスがオフライン状態になっている場合に表示されます。 デバイスファイルの障害を検知した場合は、Offline (LU Error) と表示されます。 Error 次のどれかの場合に表示されます。 <ul style="list-style-type: none"> <li>・ FC パスに障害が発生しているとき</li> <li>・ 対象の FC パスに属する全デバイスファイルにアクセスできないとき</li> <li>・ FC パスに対応づけられたホストグループへ LU が割り当てられていないとき</li> </ul> Partially Online FC パスは正常に稼働していますが、一部のデバイスファイルにアクセスできない状態の場合に表示されます。 デバイスファイルの障害を検知した場合は、Partially Online (LU Error) と表示されます。 Configuration Mismatch FC パスに対応づけられたホストグループへの LU 割り当てが、交替パスの割り当てと異なる場合、または交替パスがないときに表示されます。 Unknown ノード側の FC ポート（ホストポート）またはストレージシステム側の FC ポート（ストレージポート）を特定できない。

FCPath リソースに GET メソッドをリクエストして、FC パスの情報を取得する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/FCPaths

#### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FCPaths>
  <FCPath>
    <path>path000-0004-1E</path>
    <target>N1-T000</target>
    <hostPort>fc0004</hostPort>
    <hostPortWWN>10000000c98f2992</hostPortWWN>
    <arrayPort>1E</arrayPort>
    <arrayPortWWN>50060e801046889c</arrayPortWWN>
  </FCPath>
</FCPaths>
```

```

        <serial>83016761</serial>
        <status>Online</status>
    </FCPath>
    <FCPath>
        <path>path000-0005-1F</path>
        <target>N1-T000</target>
        <hostPort>fc0005</hostPort>
        <hostPortWWN>10000000c98f2993</hostPortWWN>
        <arrayPort>1F</arrayPort>
        <arrayPortWWN>50060e801046889d</arrayPortWWN>
        <serial>83016761</serial>
        <status>Online</status>
    </FCPath>
</FCPaths>

```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-26：FCPath リソースに GET メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3：」を参照してください。

## 3.5 FileSystem

### 3.5.1 FileSystem リソースの概要

FileSystem リソースを使用して、ファイルシステムの情報を取得したり、ファイルシステムを作成したりできます。また、ファイルシステムを削除したり、ファイルシステムが存在するかどうかを確認したりできます。

FileSystem リソースに使用できる HTTP メソッドを次の表に示します。

**表 3-27：FileSystem リソースに使用できる HTTP メソッド**

リソースの URI	HTTP メソッド	サポート構成	説明
/FileSystems	GET	クラスタ Virtual Server	ノード上または Virtual Server 上の全ファイルシステムの情報を取得します。
	PUT	クラスタ Virtual Server	ファイルシステムを作成します。
/FileSystems/ <ファイルシステム名>	GET	クラスタ Virtual Server	指定したファイルシステムの情報を取得します。
	POST	クラスタ Virtual Server	指定したファイルシステムの情報を編集します。
	DELETE	クラスタ Virtual Server	指定したファイルシステムを削除します。
	HEAD	クラスタ Virtual Server	指定したファイルシステムが存在するかどうかを確認します。
/FileSystems/ <ファイルシステム名> / MountSetting	GET	クラスタ Virtual Server	指定したファイルシステムのマウント設定の情報を取得します。
	POST	クラスタ Virtual Server	指定したファイルシステムのマウント設定の情報を編集します。

### 3.5.2 FileSystem リソース

この項では、FileSystem リソースの使い方について説明します。

### 3.5.2.1

## GET メソッドでファイルシステムの情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

表 3-28 : FileSystem リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ			出力される個数	サポート構成	verbose が false の場合の表示
FileSystems			1	クラスタ Virtual Server	○
--	FileSystem		0, 1 ~ n	クラスタ Virtual Server	○
	--	fileName	1	クラスタ Virtual Server	○
		usage	1	クラスタ Virtual Server	×
		deviceStatus	1	クラスタ Virtual Server	×
		freeBlocksInMegaBytes	1	クラスタ Virtual Server	×
		usedBlocksInMegaBytes	1	クラスタ Virtual Server	×
		totalBlocksInMegaBytes	1	クラスタ Virtual Server	×
		isLvmEnabled	1	クラスタ Virtual Server	○
		fileSystemSizeInGigaByte	1	クラスタ Virtual Server	×
		isLvmStripingEnabled	1	クラスタ Virtual Server	○
		stripes	1	クラスタ Virtual Server	×
		stripeSize	1	クラスタ Virtual Server	○
		aclType	1	クラスタ Virtual Server	○
		NamespaceShareSettings	0, 1	クラスタ Virtual Server	○
	--	type	0, 1	クラスタ Virtual Server	○
		FQDN	0, 1	クラスタ Virtual Server	○
		user	0, 1	クラスタ Virtual Server	○
		password	0, 1	クラスタ Virtual Server	○ (空文字)
	UseVersioning		0, 1	クラスタ Virtual Server	○
	periodToHold		0, 1	クラスタ Virtual Server	○
	isBypassEnabled		1	クラスタ Virtual Server	○

プロパティ				出力される個数	サポート構成	verbose が false の場合の表示
		WormSetting		1	クラスタ Virtual Server	○
		--	maxRetention	1	クラスタ Virtual Server	○
			minRetention	1	クラスタ Virtual Server	○
			isAutoCommitEnabled	1	クラスタ Virtual Server	○
			commitModeSetting	1	クラスタ Virtual Server	○
			autoCommitPeriod	1	クラスタ Virtual Server	○
			defaultRetention	1	クラスタ Virtual Server	○
			isWormDeleteEnabled	1	クラスタ Virtual Server	×
			longestRetention	1	クラスタ Virtual Server	×
		MountSetting		1	クラスタ Virtual Server	×
		--	isMounted	1	クラスタ Virtual Server	×
			mountStatus	1	クラスタ Virtual Server	×
			isReadOnly	1	クラスタ Virtual Server	×
			isFileLastAccessTimeRecordingEnabled	1	クラスタ Virtual Server	×
		LUs		1	クラスタ Virtual Server	○
		--	LU	1 ~ n	クラスタ Virtual Server	○
			-- deviceFileName	1	クラスタ Virtual Server	○

(凡例) n : 2 以上の整数を示します。○ : 表示されます。× : 表示されません。

FileSystem リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

表 3-29 : FileSystem リソースに GET メソッドをリクエストしたときのプロパティ

プロパティ			説明
FileSystems			--
--	FileSystem		--
	--	fileName	ファイルシステム名が表示されます。

プロパティ			説明
		usage	<p>ファイルシステムを運用している機能の名称が表示されます。</p> <p>None 次に示すどの機能でも運用されていない場合に表示されます。</p> <p>File snapshots ファイルスナップショット機能で運用されている場合に表示されます。</p> <p>HFRR File Remote Replicator の機能で運用されている場合に表示されます。</p> <p>WORM WORM 対応ファイルシステムの場合に表示されます。</p> <p>Snapshot 差分スナップショットである場合に表示されます。</p> <p>HFRR-Snapshot File Remote Replicator で使用されている差分スナップショットである場合に表示されます。</p> <p>なお、上記の項目のうち複数該当する場合は、各項目が (, ) で区切られて表示されます。</p>
		deviceStatus	<p>ファイルシステムを構成しているデバイスファイルの状態が表示されます。</p> <p>Normal 正常な場合に表示されます。</p> <p>Error デバイスファイルへのアクセス中に障害を検出した場合に表示されます。</p>
		freeBlocksInMegaBytes	<p>ファイルシステムが正常にマウントされている場合、ブロックの残容量が MB 単位で表示されます。</p> <p>小数点第 4 位以下は切り捨てて表示されます。</p> <p>正常にマウントされていない場合は、空文字列が表示されます。</p>
		usedBlocksInMegaBytes	<p>ファイルシステムが正常にマウントされている場合、ブロックの使用量が MB 単位で表示されます。</p> <p>小数点第 4 位以下は切り捨てて表示されます。</p> <p>正常にマウントされていない場合は、空文字列が表示されます。</p>
		totalBlocksInMegaBytes	<p>ファイルシステムが正常にマウントされている場合、ブロック容量が MB 単位で表示されます。</p> <p>小数点第 4 位以下は切り捨てて表示されます。</p> <p>正常にマウントされていない場合は、空文字列が表示されます。</p>
		isLvmEnabled	<p>ファイルシステムがボリュームマネージャーを使用しているかどうかが表示されます。</p> <p>true ボリュームマネージャーを使用している場合に表示されます。</p> <p>false ボリュームマネージャーを使用していない場合に表示されます。</p>
		volumeGroupName	<p>ファイルシステムで使用しているボリュームグループの名称が表示されます。</p>

プロパティ			説明
		fileSystemSizeInGigaByte	ファイルシステムの容量（デバイスファイルの総容量）が GB 単位で表示されます。 小数点第 4 位以下は切り捨てて表示されます。 差分スナップショットの場合は、空文字列が表示されます。
		isLvmStripingEnabled	ファイルシステムがストライピング構成で構築されているかどうかが表示されます。 true ファイルシステムがストライピング構成の場合に表示されます。 false ファイルシステムがストライピング構成ではない場合に表示されます。
		stripes	ストライプ数が整数で表示されます。 ファイルシステムがストライピング構成でない場合は、空文字列が表示されます。
		stripeSize	ストライプサイズが KB 単位で表示されます。 ファイルシステムがストライピング構成でない場合は、空文字列が表示されます。
		aclType	ファイルシステムの ACL タイプが表示されます。 Classic ACL Classic ACL タイプの場合に表示されます。 Advanced ACL Advanced ACL タイプの場合に表示されます。 Unknown ACL タイプを取得できなかった場合に表示されます。
		UseVersioning	--
		isBypassEnabled	CIFS 走査チェックのバイパス機能が有効かどうかが表示されます。 true CIFS 走査チェックのバイパス機能が有効な場合に表示されます。 false CIFS 走査チェックのバイパス機能が無効な場合に表示されます。 情報を取得できなかった場合は、空文字列が表示されます。
		WormSetting <sup>*</sup>	--
		--	
		maxRetention	WORM 対応ファイルシステムの最大リテンション期間が次の形式で表示されます。 <日>-<時>-<分> 最大リテンション期間が無期限に設定されている場合は infinite が表示されます。WORM 対応ファイルシステムでない場合は、空文字列が表示されます。
		minRetention	WORM 対応ファイルシステムの最小リテンション期間が次の形式で表示されます。 <日>-<時>-<分> 最小リテンション期間が無期限に設定されている場合は infinite が表示されます。WORM 対応ファイルシステムでない場合は、空文字列が表示されます。

プロパティ				説明
			isAutoCommitEnabled	<p>WORM 対応ファイルシステムの自動コミットが有効になっているかどうかが表示されます。</p> <p>true</p> <p>自動コミットが有効な場合に表示されます。</p> <p>false</p> <p>自動コミットが無効な場合に表示されます。</p> <p>WORM 対応ファイルシステムでない場合は、空文字列が表示されます。</p>
			commitModeSetting	<p>WORM 対応ファイルシステムの自動コミットが有効になっている場合、自動コミットのモードが表示されます。</p> <p>auto</p> <p>自動モードの場合に表示されます。</p> <p>manual</p> <p>手動モードの場合に表示されます。</p> <p>Unknown</p> <p>情報が取得できない場合に表示されます。</p> <p>WORM 対応ファイルシステムでない場合、または自動コミットが無効の場合は、空文字列が表示されます。</p>
			autoCommitPeriod	<p>WORM 対応ファイルシステムの自動コミットが有効になっている場合、ファイルを WORM 化するまでの期間が次の形式で表示されます。</p> <p>&lt;日&gt;-&lt;時&gt;-&lt;分&gt;</p> <p>WORM 対応ファイルシステムでない場合、または自動コミットが無効の場合は、空文字列が表示されます。</p>
			defaultRetention	<p>WORM 対応ファイルシステムの自動コミットが行われたファイルに設定されるリテンション期間が、次の形式で表示されます。</p> <p>&lt;日&gt;-&lt;時&gt;-&lt;分&gt;</p> <p>デフォルトリテンション期間が無期限に設定されている場合は infinite が表示されます。WORM 対応ファイルシステムでない場合、または自動コミットが無効の場合は、空文字列が表示されます。</p>
			isWormDeleteEnabled	<p>WORM 対応ファイルシステムを削除できるかどうかが表示されます。</p> <p>true</p> <p>WORM 対応ファイルシステムを削除できる場合に表示されます。</p> <p>false</p> <p>WORM 対応ファイルシステムを削除できない場合に表示されます。</p>
			longestRetention	<p>WORM 対応ファイルシステム内のファイルの最長のリテンション期間が 10 進数（秒単位）で表示されます。</p> <p>次の場合は、空文字列が表示されます。</p> <ul style="list-style-type: none"> <li>isWormDeleteEnabled プロパティが true の場合</li> <li>WORM 対応ファイルシステムではない場合</li> <li>WORM 対応ファイルシステムが閉塞状態の場合</li> </ul>
		MountSetting		--
	--		isMounted	<p>ファイルシステムがマウントされているかどうかが表示されます。</p> <p>true</p> <p>マウントされている場合に表示されます。</p> <p>false</p> <p>マウントされていない場合に表示されます。</p>

プロパティ			説明
		mountStatus	<p>ファイルシステムのマウント状態が表示されます。</p> <p>Normal 正常な場合に表示されます。</p> <p>Fatal error 閉塞状態の場合に表示されます。</p> <p>Overflow 差分格納デバイスの容量が不足した場合に表示されます。</p> <p>Not available 差分格納デバイスに障害が発生している場合に表示されます。</p> <p>Blocked 差分格納デバイスの容量の不足によってファイルシステムがブロックされている場合に表示されます。</p> <p>Blocked and Ready 差分格納デバイスの容量の不足によってファイルシステムがブロックされた際の対処として、差分格納デバイスの空き容量を確保したあと、ファイルシステムの回復処理が完了していない場合に表示されます。</p> <p>差分格納デバイスの容量不足によってファイルシステムがブロックされた際の対処については、「トラブルシューティングガイド」を参照してください。</p> <p>Expanding ファイルシステムを拡張する処理を実行中か、処理でエラーが発生している場合に表示されます。</p> <p>Reclaiming ファイルシステムで使用している仮想 LU の未使用領域を解放中の場合に表示されます。</p> <p>マウントされていない場合は、空文字列が表示されます。</p>
		isReadOnly	<p>ファイルシステムが読み取り専用でマウントされているかどうかが表示されます。</p> <p>true 読み取り専用でマウントされている場合に表示されます。</p> <p>false 読み取りと書き込みが許可された状態でマウントされている場合に表示されます。</p> <p>マウントされていない場合は、空文字列が表示されます。</p>
		isFileLastAccessTimeRecordingEnabled	<p>ファイルシステムにアクセスしたときの最終アクセス時刻が更新されるように設定されているかどうかが表示されます。</p> <p>true 最終アクセス時刻が更新されるように設定されている場合に表示されます。</p> <p>false 最終アクセス時刻が更新されるように設定されていない場合に表示されます。</p> <p>マウントされていない場合は、空文字列が表示されます。</p>
		LUs	--



プロパティ				説明
		--	LU	--
		--	deviceFileName	ファイルシステムで使用しているデバイスファイルの名称が表示されます。isLvmStripingEnabled プロパティに true が指定されている場合は、ファイルシステムを構築または拡張したときに指定した順で表示されます。その他の場合は、昇順で表示されます。

注 \* マウントされていない場合、値は表示されません。

FileSystem リソースに GET メソッドをリクエストして、ファイルシステムの情報を取得する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems

### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FileSystems>
  <FileSystem>
    <fileName>fs01</fileName>
    <usage>None</usage>
    <deviceStatus>Normal</deviceStatus>
    <freeBlocksInMegaBytes>15345.664</freeBlocksInMegaBytes>
    <usedBlocksInMegaBytes>4.096</usedBlocksInMegaBytes>
    <totalBlocksInMegaBytes>15349.76</totalBlocksInMegaBytes>
    <fileSystemSizeInGigaByte>15.000</fileSystemSizeInGigaByte>
    <aclType>Advanced ACL</aclType>
    <NamespaceShareSettings>
      <type>Read/Write</type>
      <FQDN></FQDN>
      <user></user>
      <password></password>
    </NamespaceShareSettings>
    <UseVersioning>used</UseVersioning>
    <periodToHold>7</periodToHold>
    <isBypassEnabled>true</isBypassEnabled>
    <WormSetting>
      <maxRetention></maxRetention>
      <minRetention></minRetention>
      <isAutoCommitEnabled></isAutoCommitEnabled>
      <commitModeSetting></commitModeSetting>
      <autoCommitPeriod></autoCommitPeriod>
      <defaultRetention></defaultRetention>
      <isWormDeleteEnabled></isWormDeleteEnabled>
      <longestRetention></longestRetention>
    </WormSetting>
    <MountSetting>
      <isMounted>true</isMounted>
      <mountStatus>Normal</mountStatus>
      <isReadOnly>false</isReadOnly>
      <isFileLastAccessTimeRecordingEnabled>true</isFileLastAccessTimeRecordingEnabled>
    </MountSetting>
  </FileSystem>
</FileSystems>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-30 : FileSystem リソースに GET メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.5.2.2

### PUT メソッドでファイルシステムを作成する

FileSystem リソースに PUT メソッドをリクエストするときの XML の構造を次の表に示します。

表 3-31 : FileSystem リソースに PUT メソッドをリクエストするときの XML の構造

プロパティ			データ型	指定できる個数	サポート構成
FileSystem			--	1	クラスタ Virtual Server
--	fileName		String	1	クラスタ Virtual Server
	isLvmEnabled		Boolean	0, 1	クラスタ Virtual Server
	isLvmStripingEnabled		Boolean	0, 1	クラスタ Virtual Server
	stripeSize		Integer	0, 1	クラスタ Virtual Server
	aclType		String	0, 1	クラスタ Virtual Server
	NamespaceShareSettings		--	0, 1	クラスタ Virtual Server
	--	type	String	0, 1	クラスタ Virtual Server
		FQDN	String	0, 1	クラスタ Virtual Server
		user	String	0, 1	クラスタ Virtual Server
		password	String	0, 1	クラスタ Virtual Server
	UseVersioning		String	0, 1	クラスタ Virtual Server
	periodToHold		Integer	0, 1	クラスタ Virtual Server
	isBypassEnabled		Boolean	0, 1	クラスタ Virtual Server
	WormSetting		--	0, 1	クラスタ Virtual Server
	--	maxRetention	String	0, 1	クラスタ Virtual Server
		minRetention	String	0, 1	クラスタ Virtual Server
		isAutoCommitEnabled	Boolean	0, 1	クラスタ Virtual Server
commitModeSetting		String	0, 1	クラスタ Virtual Server	
autoCommitPeriod		String	0, 1	クラスタ Virtual Server	
defaultRetention		String	0, 1	クラスタ Virtual Server	
LUs		--	1	クラスタ Virtual Server	
--	LU	--	1 ～ n	クラスタ Virtual Server	
	--	deviceFileName	String	1	クラスタ Virtual Server

(凡例) n : 2 以上の整数を示します。

FileSystem リソースに PUT メソッドをリクエストするときに指定するプロパティについて次の表に示します。

表 3-32 : FileSystem リソースに PUT メソッドをリクエストするときのプロパティ

プロパティ		説明	指定時の条件
FileSystem		--	--
--	fileName	作成するファイルシステムの名称を指定します。クラスタ内で重複しない名称を指定します。 16 文字以内の英数字およびアンダーライン ( _ ) で指定します。 なお、Virtual Server 上で作成する場合は、Virtual Server 内で重複しない名称を指定します。	必須
	isLvmEnabled	ボリュームマネージャーを使用するかどうかを設定します。 true*1 ボリュームマネージャーを使用する場合に指定します。 複数のデバイスファイルからファイルシステムを構築する場合は、true を指定してください。 false ボリュームマネージャーを使用しない場合に指定します。 false を指定した場合、ファイルシステムの作成後に容量を拡張できません。 また、isLvmStripingEnabled プロパティおよび stripeSize プロパティを指定できません。	任意
	isLvmStripingEnabled	ストライピング構成でファイルシステムを構築するかどうかを設定します。 true ストライピング構成でファイルシステムを構築する場合に指定します。 false*1 ストライピング構成でファイルシステムを構築しない場合に指定します。	任意

プロパティ		説明	指定時の条件
	stripeSize	ストライプサイズを KB 単位で設定します。 次のどれかの値を指定します。 <ul style="list-style-type: none"> <li>• 8</li> <li>• 16</li> <li>• 32</li> <li>• 64<sup>*1</sup></li> <li>• 128</li> <li>• 256</li> <li>• 512</li> <li>• 1024</li> </ul>	isLvmStripingEnabled プロパティが true の場合に指定できます。
	aclType	ファイルシステムの ACL タイプを設定します。 Classic ACL <sup>*1</sup> Classic ACL タイプのファイルシステムを構築する場合に指定します。 Advanced ACL Advanced ACL タイプのファイルシステムを構築する場合に指定します。	任意
	NamespaceShareSettings	--	任意
	UseVersioning	--	任意
	isBypassEnabled	CIFS 走査チェックのバイパス機能を有効にするかどうかを指定します。 true <sup>*1</sup> CIFS 走査チェックのバイパス機能を有効にする場合に指定します。 false CIFS 走査チェックのバイパス機能を無効にする場合に指定します。	type プロパティが HomeDirectory の場合は指定できません。
	WormSetting	WORM 対応ファイルシステムを構築する場合に指定します。 WORM 機能の有効・無効の設定は、ファイルシステムの構築後に変更できません。	任意

プロパティ			説明	指定時の条件
	--	maxRetention	<p>WORM 対応ファイルシステム の最大リテンション期間を設定 します。</p> <p>infinite または次の形式で設 定します。</p> <p>&lt;日&gt;-&lt;時&gt;-&lt;分&gt; 設定できる値は、次のとおりで す。</p> <ul style="list-style-type: none"> <li>・ &lt;日&gt; : 0 ~ 36500</li> <li>・ &lt;時&gt; : 0 ~ 876000</li> <li>・ &lt;分&gt; : 0 ~ 52560000</li> </ul> <p>設定できる期間は、1 分 (0-0- 1) ~ 100 年 (36500-0-0) で す。<sup>*2</sup></p> <p>infinite を設定した場合は、 最大リテンション期間が無期限 になります。</p> <p>省略した場合は、10950-0-0 が設定されます。</p>	WormSetting プロパティ を設定する場合に指定でき ます。
		minRetention	<p>WORM 対応ファイルシステム の最小リテンション期間を設定 します。</p> <p>infinite または次の形式で設 定します。</p> <p>&lt;日&gt;-&lt;時&gt;-&lt;分&gt; 設定できる値は、次のとおりで す。</p> <ul style="list-style-type: none"> <li>・ &lt;日&gt; : 0 ~ 36500</li> <li>・ &lt;時&gt; : 0 ~ 876000</li> <li>・ &lt;分&gt; : 0 ~ 52560000</li> </ul> <p>設定できる期間は、0 分 (0-0- 0) ~ 100 年 (36500-0-0) で す。<sup>*2</sup></p> <p>infinite を設定した場合は、 最小リテンション期間が無期限 になります。</p> <p>省略した場合は、0-0-0 が設定 されます。</p>	WormSetting プロパティ を設定する場合に指定でき ます。
		isAutoCommitEnabled	<p>WORM 対応ファイルシステム で自動コミットを有効にするか どうかを指定します。</p> <p>true から false には変更でき ません。</p> <p>true</p> <p>自動コミットを有効にする 場合に指定します。</p> <p>false<sup>*1</sup></p> <p>自動コミットを無効にする 場合に指定します。</p>	WormSetting プロパティ を設定する場合に指定でき ます。

プロパティ		説明	指定時の条件
	commitModeSetting	<p>WORM 対応ファイルシステム の自動コミットのモードを指定 します。</p> <p>auto</p> <p>自動コミットを自動モード で有効にする場合に指定し ます。</p> <p>システムファイルおよびシ ステムディレクトリ以下の ファイルを除く、すべての 通常ファイルが自動コミッ トの対象となります。</p> <p>manual<sup>*1</sup></p> <p>自動コミットを手動モード で有効にする場合に指定し ます。</p> <p>クライアントから読み取り 専用にしたファイルが自動 コミットの対象となりま す。</p>	<p>次の条件をすべて満たす場 合に指定できます。</p> <ul style="list-style-type: none"> <li>WormSetting プロパ ティを設定する</li> <li>isAutoCommitEnable d プロパティが true である</li> </ul>
	autoCommitPeriod	<p>ファイルが WORM 化するまで の期間を設定します。一度設定 すると変更できません。</p> <p>次の形式で設定します。</p> <p>&lt;日&gt;-&lt;時&gt;-&lt;分&gt; 設定できる値は、次のとおりで す。</p> <ul style="list-style-type: none"> <li>&lt;日&gt; : 0 ~ 36500</li> <li>&lt;時&gt; : 0 ~ 876000</li> <li>&lt;分&gt; : 0 ~ 52560000</li> </ul> <p>設定できる期間は、1 分 (0-0- 1) ~ 100 年 (36500-0-0) で す。<sup>*2</sup></p> <p>省略した場合は、0-0-15 が設 定されます。</p>	<p>次の条件をすべて満たす場 合に指定できます。</p> <ul style="list-style-type: none"> <li>WormSetting プロパ ティを設定する</li> <li>isAutoCommitEnable d プロパティが true である</li> </ul>
	defaultRetention	<p>自動コミットされたファイルの デフォルトリテンション期間を 設定します。</p> <p>infinite または次の形式で設 定します。</p> <p>&lt;日&gt;-&lt;時&gt;-&lt;分&gt; 設定できる値は、次のとおりで す。</p> <ul style="list-style-type: none"> <li>&lt;日&gt; : 0 ~ 36500</li> <li>&lt;時&gt; : 0 ~ 876000</li> <li>&lt;分&gt; : 0 ~ 52560000</li> </ul> <p>設定できる期間は、1 分 (0-0- 1) ~ 100 年 (36500-0-0) で す。<sup>*2</sup></p> <p>infinite を設定した場合は、 デフォルトリテンション期間が 無期限になります。</p> <p>省略した場合は、10-0-0 が設 定されます。ただし、 minRetention プロパティに infinite を設定している場合 は infinite が設定されます。</p>	<p>次の条件をすべて満たす場 合に指定できます。</p> <ul style="list-style-type: none"> <li>WormSetting プロパ ティを設定する</li> <li>isAutoCommitEnable d プロパティが true である</li> </ul>

プロパティ				説明	指定時の条件
	LUs			--	--
	--	LU		--	--
		--	deviceFileName	<p>ファイルシステムに割り当てるデバイスファイルの名称を次の形式で指定します。</p> <p>lu &lt;ユーザー LUN &gt;</p> <p>ユーザー LUN は 16 進数で 00 ～ FF または 0000 ～ 03FF の範囲を指定できます。</p> <p>isLvmEnabled プロパティに true を指定した場合は、65MB 以上のデバイスファイルを指定してください。また、指定するデバイスファイルの総容量が、仮想 LU を使用する場合は 256MB 以上、仮想 LU を使用しない場合は 130MB 以上になるようにしてください。ストライピング構成でファイルシステムを構築する場合は、容量がすべて同一のデバイスファイルを 2 ～ 128 個で指定してください。指定したデバイスファイルの数がストライプ数になります。また、指定したデバイスファイルの順にストライプが構成されます。</p>	必須

注 \*1 プロパティを省略した場合のデフォルト値です。

注 \*2 例えば、30-15-10 と指定した場合は、30 日 15 時間 10 分が設定されます。0-48-0 と 2-0-0 では、同じ期間（2 日間）が設定されます。

FileSystem リソースに PUT メソッドをリクエストして、ファイルシステムを作成する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems

### リクエスト XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FileSystem>
  <fileName>fs05</fileName>
  <isLvmEnabled>true</isLvmEnabled>
  <isLvmStripingEnabled>false</isLvmStripingEnabled>
  <stripeSize></stripeSize>
  <aclType>Classic ACL</aclType>
  <NamespaceShareSettings>
    <type>Read/Write</type>
  </NamespaceShareSettings>
  <UseVersioning>used</UseVersioning>
  <periodToHold>7</periodToHold>
  <isBypassEnabled>true</isBypassEnabled>
  <WormSetting>
    <maxRetention>36500-0-0</maxRetention>
    <minRetention>0-0-0</minRetention>
    <isAutoCommitEnabled>true</isAutoCommitEnabled>
    <commitModeSetting>manual</commitModeSetting>
    <autoCommitPeriod>0-0-15</autoCommitPeriod>
    <defaultRetention>10950-0-0</defaultRetention>
  </WormSetting>
</FileSystem>
```

```

</WormSetting>
<LU>
  <deviceFileName>lu0003</deviceFileName>
</LU>
<LU>
  <deviceFileName>lu0005</deviceFileName>
</LU>
</LUs>
</FileSystem>

```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-33： FileSystem リソースに PUT メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3：」を参照してください。

### 3.5.3 FileSystems/ <ファイルシステム名> リソース

この項では、FileSystems/ <ファイルシステム名> リソースの使い方について説明します。

#### 3.5.3.1 GET メソッドで指定したファイルシステムの情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

**表 3-34： FileSystems/ <ファイルシステム名> リソースに GET メソッドをリクエストしたときのレスポンス XML の構造**

プロパティ	出力される 個数	サポート構成	verbose が false の場合の表示
FileSystem	1	クラスタ Virtual Server	○



プロパティ		出力される 個数	サポート構成	verbose が false の場合の表示
--	fileSystemName	1	クラスタ Virtual Server	○
	usage	1	クラスタ Virtual Server	×
	deviceStatus	1	クラスタ Virtual Server	×
	freeBlocksInMegaBytes	1	クラスタ Virtual Server	×
	usedBlocksInMegaBytes	1	クラスタ Virtual Server	×
	totalBlocksInMegaBytes	1	クラスタ Virtual Server	×
	isLvmEnabled	1	クラスタ Virtual Server	×
	fileSystemSizeInGigaByte	1	クラスタ Virtual Server	×
	isLvmStripingEnabled	1	クラスタ Virtual Server	×
	stripes	1	クラスタ Virtual Server	×
	stripeSize	1	クラスタ Virtual Server	×
	aclType	1	クラスタ Virtual Server	○

プロパティ			出力される 個数	サポート構成	verbose が false の場合の表示	
	NamespaceShareSettings		0, 1	クラスタ Virtual Server	○	
	--	type	0, 1	クラスタ Virtual Server	○	
		FQDN	0, 1	クラスタ Virtual Server	○	
		user	0, 1	クラスタ Virtual Server	○	
		password	0, 1	クラスタ Virtual Server	○	
	UseVersioning		0, 1	クラスタ Virtual Server	○	
	periodToHold		0, 1	クラスタ Virtual Server	○	
	isBypassEnabled		1	クラスタ Virtual Server	○	
	WormSetting		1	クラスタ Virtual Server	○	
	--	maxRetention		1	クラスタ Virtual Server	○
		minRetention		1	クラスタ Virtual Server	○
		isAutoCommitEnabled		1	クラスタ Virtual Server	○
		commitModeSetting		1	クラスタ Virtual Server	○
		autoCommitPeriod		1	クラスタ Virtual Server	○
		defaultRetention		1	クラスタ Virtual Server	○
		isWormDeleteEnabled		1	クラスタ Virtual Server	×
		longestRetention		1	クラスタ Virtual Server	×
	MountSetting		1	クラスタ Virtual Server	×	
	--	isMounted		1	クラスタ Virtual Server	×
		mountStatus		1	クラスタ Virtual Server	×
isReadOnly		1	クラスタ Virtual Server	×		
isFileLastAccessTimeRecordingEnabled		1	クラスタ Virtual Server	×		
LUs		1	クラスタ Virtual Server	○		
--	LU		1 ～ n	クラスタ Virtual Server	○	
	--	deviceFileName	1	クラスタ Virtual Server	○	

(凡例) n : 2 以上の整数を示します。○ : 表示されます。× : 表示されません。

FileSystems/ <ファイルシステム名> リソースに GET メソッドをリクエストしたときに表示されるプロパティについては、「[表 3-29 : FileSystem リソースに GET メソッドをリクエストしたときのプロパティ](#)」を参照してください。

FileSystems/ <ファイルシステム名> リソースに GET メソッドをリクエストして、指定したファイルシステムの情報を取得する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems/<ファイルシステム名>

#### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FileSystem>
  <fileName>fs01</fileName>
  <usage>None</usage>
  <deviceStatus></deviceStatus>
  <freeBlocksInMegaBytes>5105.664</freeBlocksInMegaBytes>
  <usedBlocksInMegaBytes>4.096</usedBlocksInMegaBytes>
  <totalBlocksInMegaBytes>5109.76</totalBlocksInMegaBytes>
  <fileSystemSizeInGigaByte>5.000</fileSystemSizeInGigaByte>
  <aclType>Advanced ACL</aclType>
  <NamespaceShareSettings>
    <type>Read/Write</type>
    <FQDN></FQDN>
    <user></user>
    <password></password>
  </NamespaceShareSettings>
  <UseVersioning>used</UseVersioning>
  <periodToHold></periodToHold>
  <isBypassEnabled>true</isBypassEnabled>
  <WormSetting>
    <maxRetention></maxRetention>
    <minRetention></minRetention>
    <isAutoCommitEnabled></isAutoCommitEnabled>
    <commitModeSetting></commitModeSetting>
    <autoCommitPeriod></autoCommitPeriod>
    <defaultRetention></defaultRetention>
    <isWormDeleteEnabled></isWormDeleteEnabled>
    <longestRetention></longestRetention>
  </WormSetting>
  <MountSetting>
    <isMounted>true</isMounted>
    <mountStatus>Normal</mountStatus>
    <isReadOnly>false</isReadOnly>
    <isFileLastAccessTimeRecordingEnabled>true</
isFileLastAccessTimeRecordingEnabled>
  </MountSetting>
</FileSystem>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-35 : FileSystems/ <ファイルシステム名> リソースに GET メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「 <a href="#">表 1-3 :</a> 」を参照してください。

### 3.5.3.2

#### POST メソッドで指定したファイルシステムの情報を編集する

FileSystems/ <ファイルシステム名> リソースに POST メソッドをリクエストするときの XML の構造を次の表に示します。

表 3-36 : FileSystems/ <ファイルシステム名>リソースに POST メソッドをリクエストするときの XML の構造

プロパティ			データ型	指定できる個数	サポート構成
FileSystem			--	1	クラスタ Virtual Server
	NamespaceShareSettings		--	0, 1	クラスタ Virtual Server
	--	type	String	0, 1	クラスタ Virtual Server
		user	String	0, 1	クラスタ Virtual Server
		password	String	0, 1	クラスタ Virtual Server
	UseVersioning		Boolean	0, 1	クラスタ Virtual Server
	periodToHold		Integer	0, 1	クラスタ Virtual Server
	isBypassEnabled		Boolean	0, 1	クラスタ Virtual Server
	addLUs		--	0, 1	クラスタ Virtual Server
	--	LU	--	1 ~ n	クラスタ Virtual Server
		-- deviceFileName	String	1	クラスタ Virtual Server
	WormSetting		--	0, 1	クラスタ Virtual Server
	--	maxRetention	String	0, 1	クラスタ Virtual Server
		minRetention	String	0, 1	クラスタ Virtual Server
		isAutoCommitEnabled	Boolean	0, 1	クラスタ Virtual Server
		commitModeSetting	String	0, 1	クラスタ Virtual Server
		autoCommitPeriod	String	0, 1	クラスタ Virtual Server
		defaultRetention	String	0, 1	クラスタ Virtual Server

(凡例) n : 2 以上の整数を示します。

FileSystems/ <ファイルシステム名>リソースに POST メソッドをリクエストするときに指定するプロパティについて次の表に示します。

表 3-37 : FileSystems/ <ファイルシステム名>リソースに POST メソッドをリクエストするときのプロパティ

プロパティ	説明	指定時の条件
FileSystem	--	--

プロパティ			説明	指定時の条件
--	expandFileSystemSize		ファイルシステムを拡張する容量を GB 単位で設定します。追加する容量を指定してください。	次のプロパティのうち、どれか一つを必ず指定します。 <ul style="list-style-type: none"> <li>expandFileSystemSize</li> <li>addLUs</li> <li>WormSetting</li> </ul>
	NamespaceShareSettings		--	--
	UseVersioning		--	--
	isBypassEnabled		CIFS 走査チェックのバイパス機能を有効にするかどうかを指定します。 true CIFS 走査チェックのバイパス機能を有効にする場合に指定します。 false CIFS 走査チェックのバイパス機能を無効にする場合に指定します。	任意
	addLUs		ファイルシステムを拡張するために追加する LU を設定します。ファイルシステム作成時に isLvmStripingEnabled プロパティに false を指定した場合は、容量を拡張できません。isLvmStripingEnabled プロパティに true を指定した場合は、容量がすべて同一のデバイスファイルを指定してください。また、指定するデバイスファイルの数は、ファイルシステム作成時に stripes プロパティで設定した数と同じにしてください。	次のプロパティのうち、どれか一つを必ず指定します。 <ul style="list-style-type: none"> <li>expandFileSystemSize</li> <li>addLUs</li> <li>WormSetting</li> </ul>
	--	LU	ファイルシステムに割り当てる LU を設定します。	addLUs プロパティを設定する場合に必須です。
		deviceFileName	ファイルシステムを割り当てるデバイスファイルの名称を次の形式で指定します。 lu <ユーザー LUN > ユーザー LUN は 16 進数で 00 ～ FF または 0000 ～ 01FF の範囲を指定できます。	addLUs プロパティを設定する場合に必須です。
	WormSetting		--	次のプロパティのうち、どれか一つを必ず指定します。 <ul style="list-style-type: none"> <li>expandFileSystemSize</li> <li>addLUs</li> <li>WormSetting</li> </ul>

プロパティ		説明	指定時の条件
	--	<p>maxRetention</p> <p>WORM 対応ファイルシステムの最大リテンション期間を設定します。</p> <p>infinite または次の形式で設定します。</p> <p>&lt;日&gt;-&lt;時&gt;-&lt;分&gt;</p> <p>設定できる値は、次のとおりです。</p> <ul style="list-style-type: none"> <li>• &lt;日&gt; : 0 ~ 36500</li> <li>• &lt;時&gt; : 0 ~ 876000</li> <li>• &lt;分&gt; : 0 ~ 52560000</li> </ul> <p>設定できる期間は、1 分 (0-0-1) ~ 100 年 (36500-0-0) です。* infinite を設定した場合は、最大リテンション期間が無期限になります。</p>	<p>次の条件をすべて満たす場合に指定できます。</p> <ul style="list-style-type: none"> <li>• WormSetting プロパティを設定する</li> <li>• minRetention プロパティが infinite ではない</li> </ul>
		<p>minRetention</p> <p>WORM 対応ファイルシステムの最小リテンション期間を設定します。</p> <p>infinite または次の形式で設定します。</p> <p>&lt;日&gt;-&lt;時&gt;-&lt;分&gt;</p> <p>設定できる値は、次のとおりです。</p> <ul style="list-style-type: none"> <li>• &lt;日&gt; : 0 ~ 36500</li> <li>• &lt;時&gt; : 0 ~ 876000</li> <li>• &lt;分&gt; : 0 ~ 52560000</li> </ul> <p>設定できる期間は、0 分 (0-0-0) ~ 100 年 (36500-0-0) です。* infinite を設定した場合は、最小リテンション期間が無期限になります。</p>	<p>WormSetting プロパティを設定する場合に指定できます。</p>
		<p>isAutoCommitEnabled</p> <p>WORM 対応ファイルシステムで自動コミットを有効にするかどうかを指定します。いったん有効にすると、無効に変更することはできなくなります。</p> <p>true</p> <p>自動コミットを有効にする場合に指定します。</p>	<p>WormSetting プロパティを設定する場合に指定できます。</p>

プロパティ		説明	指定時の条件
	commitModeSetting	<p><b>WORM</b> 対応ファイルシステムの自動コミットのモードを指定します。</p> <p>auto</p> <p>自動コミットを自動モードで有効にする場合に指定します。</p> <p>システムファイルおよびシステムディレクトリ以下のファイルを除く、すべての通常ファイルが自動コミットの対象となります。</p> <p>manual</p> <p>自動コミットを手動モードで有効にする場合に指定します。</p> <p>クライアントから読み取り専用にしたファイルが自動コミットの対象となります。</p>	<p>次の条件をすべて満たす場合に必須です。</p> <ul style="list-style-type: none"> <li>WormSetting プロパティを設定する</li> <li>isAutoCommitEnabled プロパティが true である</li> </ul>
	autoCommitPeriod	<p>ファイルが <b>WORM</b> 化するまでの期間を設定します。一度設定すると変更できません。</p> <p>次の形式で設定します。</p> <p>&lt;日&gt;-&lt;時&gt;-&lt;分&gt;</p> <p>設定できる値は、次のとおりです。</p> <ul style="list-style-type: none"> <li>&lt;日&gt; : 0 ~ 36500</li> <li>&lt;時&gt; : 0 ~ 876000</li> <li>&lt;分&gt; : 0 ~ 52560000</li> </ul> <p>設定できる期間は、1 分 (0-0-1) ~ 100 年 (36500-0-0) です。*</p>	<p>次の条件をすべて満たす場合に指定できます。</p> <ul style="list-style-type: none"> <li>WormSetting プロパティを設定する</li> <li>isAutoCommitEnabled プロパティが true である</li> </ul>
	defaultRetention	<p>自動コミットされたファイルのデフォルトリテンション期間を設定します。</p> <p>infinite または次の形式で設定します。</p> <p>&lt;日&gt;-&lt;時&gt;-&lt;分&gt;</p> <p>設定できる値は、次のとおりです。</p> <ul style="list-style-type: none"> <li>&lt;日&gt; : 0 ~ 36500</li> <li>&lt;時&gt; : 0 ~ 876000</li> <li>&lt;分&gt; : 0 ~ 52560000</li> </ul> <p>設定できる期間は、1 分 (0-0-1) ~ 100 年 (36500-0-0) です。*</p> <p>infinite を設定した場合は、デフォルトリテンション期間が無期限になります。</p>	<p>次の条件をすべて満たす場合に指定できます。</p> <ul style="list-style-type: none"> <li>WormSetting プロパティを設定する</li> <li>isAutoCommitEnabled プロパティが true である</li> </ul>

注 \* 例えば、30-15-10 と指定した場合は、30 日 15 時間 10 分が設定されます。0-48-0 と 2-0-0 では、同じ期間（2 日間）が設定されます。

FileSystems/<ファイルシステム名>リソースに POST メソッドをリクエストして、指定したファイルシステムの情報を編集する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems/<ファイルシステム名>

### リクエスト XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FileSystem>
  <addLUs>
    <LU>
      <deviceFileName>lu19</deviceFileName>
    </LU>
  </addLUs>
</FileSystem>
```

表 3-38 : FileSystems/ <ファイルシステム名> リソースに POST メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

#### 3.5.3.3

### DELETE メソッドで指定したファイルシステムを削除する

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-39 : FileSystems/ <ファイルシステム名> リソースに DELETE メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

#### 3.5.3.4

### HEAD メソッドで指定したファイルシステムが存在するかどうかを確認する

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-40 : FileSystems/ <ファイルシステム名> リソースに HEAD メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	指定したファイルシステムが存在します。
404	指定したファイルシステムが存在しません。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.5.4

## FileSystems/ <ファイルシステム名> /MountSetting リソース

この項では、FileSystems/ <ファイルシステム名> /MountSetting リソースの使い方について説明します。

#### 3.5.4.1

### GET メソッドで指定したファイルシステムのマウント設定の情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

表 3-41 : FileSystems/ <ファイルシステム名> /MountSetting リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ	出力される個数	サポート構成	verbose が false の場合の表示
MountSetting	1	クラスタ Virtual Server	○



プロパティ		出力される個数	サポート構成	verbose が false の場合の表示
--	isMounted	1	クラスタ Virtual Server	○
	isReadOnly	1	クラスタ Virtual Server	○
	isFileLastAccessTimeRecordingEnabled	1	クラスタ Virtual Server	○
	mountStatus	1	クラスタ Virtual Server	×

(凡例) ○ : 表示されます。× : 表示されません。

FileSystems/ <ファイルシステム名> /MountSetting リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

**表 3-42 : FileSystems/ <ファイルシステム名> /MountSetting リソースに GET メソッドをリクエストしたときのプロパティ**

プロパティ		説明
MountSetting		--
--	isMounted	ファイルシステムがマウントされているかどうかが表示されます。 true マウントされている場合に表示されます。 false マウントされていない場合に表示されます。

プロパティ		説明
	isReadOnly	<p>ファイルシステムが読み取り専用でマウントされているかどうかが表示されます。</p> <p>true</p> <p>読み取り専用でマウントされている場合に表示されます。</p> <p>false</p> <p>読み取りと書き込みが許可された状態でマウントされている場合に表示されます。</p> <p>マウントされていない場合は、空文字列が表示されます。</p>
	isFileLastAccessTimeRecordingEnabled	<p>ファイルシステムにアクセスしたときの最終アクセス時刻が更新されるように設定されているかどうかが表示されます。</p> <p>true</p> <p>最終アクセス時刻が更新されるように設定されている場合に表示されます。</p> <p>false</p> <p>最終アクセス時刻が更新されるように設定されていない場合に表示されます。</p> <p>マウントされていない場合は、空文字列が表示されます。</p>
	mountStatus	<p>ファイルシステムのマウント状態が表示されます。</p> <p>Normal</p> <p>正常な場合に表示されます。</p> <p>Fatal error</p> <p>閉塞状態の場合に表示されます。</p> <p>Overflow</p> <p>差分格納デバイスの容量が不足した場合に表示されます。</p> <p>Not available</p> <p>差分格納デバイスに障害が発生している場合に表示されます。</p> <p>Blocked</p> <p>差分格納デバイスの容量の不足によってファイルシステムがブロックされている場合に表示されます。</p> <p>Blocked and Ready</p> <p>差分格納デバイスの容量の不足によってファイルシステムがブロックされた際の対処として、差分格納デバイスの空き容量を確保したあと、ファイルシステムの回復処理が完了していない場合に表示されます。</p> <p>差分格納デバイスの容量不足によってファイルシステムがブロックされた際の対処については、「トラブルシューティングガイド」を参照してください。</p> <p>Expanding</p> <p>ファイルシステムを拡張する処理を実行中か、処理でエラーが発生している場合に表示されます。</p> <p>Reclaiming</p> <p>ファイルシステムで使用している仮想 LU の未使用領域を解放中の場合に表示されます。</p> <p>マウントされていない場合は、空文字列が表示されます。</p>

FileSystems/<ファイルシステム名>/MountSetting リソースに GET メソッドをリクエストして、指定したファイルシステムの情報を取得する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems/<ファイルシステム名>/MountSetting

#### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<MountSetting>
  <isMounted>false</isMounted>
  <isReadOnly></isReadOnly>
  <isFileLastAccessTimeRecordingEnabled></
```

```
isFileLastAccessTimeRecordingEnabled>
</MountSetting>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-43 : FileSystems/ <ファイルシステム名> /MountSetting リソースに GET メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.5.4.2 POST メソッドで指定したファイルシステムのマウント設定の情報を編集する

FileSystems/ <ファイルシステム名> /MountSetting リソースに POST メソッドをリクエストするときの XML の構造を次の表に示します。

**表 3-44 : FileSystems/ <ファイルシステム名> /MountSetting リソースに POST メソッドをリクエストするときの XML の構造**

プロパティ		データ型	指定できる個数	サポート構成
MountSetting		--	1	クラスタ Virtual Server
--	isMounted	Boolean	1	クラスタ Virtual Server
	isReadOnly	Boolean	0, 1	クラスタ Virtual Server
	isFileLastAccessTimeRecordingEnabled	Boolean	0, 1	クラスタ Virtual Server
	isConvertToAdvancedAclType	Boolean	0, 1	クラスタ Virtual Server

FileSystems/ <ファイルシステム名> /MountSetting リソースに POST メソッドをリクエストするときに指定するプロパティについて次の表に示します。

**表 3-45 : FileSystems/ <ファイルシステム名> /MountSetting リソースに POST メソッドをリクエストするときのプロパティ**

プロパティ		説明	指定時の条件
MountSetting		--	--
--	isMounted	ファイルシステムのマウント状態を変更するかどうかを設定します。 true マウントする場合に指定します。 false*1 アンマウントする場合に指定します。	必須
	isReadOnly	ファイルシステムへの書き込みを許可するかどうかを設定します。 true 読み取り専用でマウントする場合に指定します。 false 読み取りと書き込みを許可した状態でマウントする場合に指定します。	isMounted プロパティが true の場合に必須です。

プロパティ		説明	指定時の条件
	isFileLastAccessTimeRecordingEnabled	ファイルシステムにアクセスしたときの最終アクセス時刻を更新するかどうかを設定します。 true 最終アクセス時刻を更新する場合に指定します。 false <sup>*2</sup> 最終アクセス時刻を更新しない場合に指定します。	次の条件をすべて満たす場合に指定できます。 <ul style="list-style-type: none"> <li>isMounted プロパティが true である</li> <li>isReadOnly プロパティが false である</li> </ul>
	isConvertToAdvancedAclType	ファイルシステムの ACL タイプを Classic ACL タイプから Advanced ACL タイプに変更するかどうかを設定します。変更する場合、ファイルシステム内の既存のディレクトリおよびファイルの ACL タイプは、バックグラウンドで変換されます。 true Classic ACL タイプから Advanced ACL タイプに変更する場合に指定します。 false <sup>*2</sup> Classic ACL タイプから Advanced ACL タイプに変更しない場合に指定します。	次の条件をすべて満たす場合に指定できます。 <ul style="list-style-type: none"> <li>isMounted プロパティが true である</li> <li>isReadOnly プロパティが false である</li> <li>ファイルシステムの ACL タイプが Classic ACL タイプである</li> </ul>

注 \*1 アンマウントの処理に失敗した場合、Backup Restore のボリュームレプリケーション連携機能によって、ファイルシステムに対するアクセスが一時的に抑止されているおそれがあります。fsctl コマンドでファイルシステムに対するアクセスの抑止状況を確認し、Backup Restore の horcunfreeze コマンドですべてのファイルシステムの抑止を解除してから、メソッドを再実行してください。

注 \*2 プロパティを省略した場合のデフォルト値です。

FileSystems/ <ファイルシステム名> /MountSetting リソースに POST メソッドをリクエストして、指定したファイルシステムの情報を編集する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems/<ファイルシステム名>/MountSetting

### リクエスト XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<MountSetting>
  <isMounted>true</isMounted>
  <isReadOnly>>false</isReadOnly>
  <isFileLastAccessTimeRecordingEnabled>true</isFileLastAccessTimeRecordingEnabled>
</MountSetting>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-46 : FileSystems/ <ファイルシステム名> /MountSetting リソースに POST メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

## 3.6 Hardware

### 3.6.1 Hardware リソースの概要

Hardware リソースを使用して、ハードウェアの状態を取得できます。

Hardware リソースに使用できる HTTP メソッドを次の表に示します。

表 3-47: Hardware リソースに使用できる HTTP メソッド

リソースの URI	HTTP メソッド	サポート構成	説明
/Hardware	GET	クラスタ	ログインしたノード上のハードウェアの状態を取得します。

### 3.6.2 Hardware リソース

この項では、Hardware リソースの使い方について説明します。

#### 3.6.2.1 GET メソッドでハードウェア情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

表 3-48: Hardware リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ			出力される個数	サポート構成	verbose が false の場合の表示
Hardware			1	クラスタ	○
--	PortDetails		1	クラスタ	○
	--	PortDetail	0, 1 ~ n	クラスタ	○
		status	1	クラスタ	○
		portName	1	クラスタ	○
		mediaType	1	クラスタ	○
		linkSpeed	1	クラスタ	○
	ManagementLans		1	クラスタ	○
	--	ManagementLan	0, 1	クラスタ	○
		status	1	クラスタ	○
		portName	1	クラスタ	○
		mediaType	1	クラスタ	○
		linkSpeed	1	クラスタ	○
	HeartbeatLans		0, 1	クラスタ	○
	--	HeartbeatLan	0, 1	クラスタ	○
		status	1	クラスタ	○
		portName	1	クラスタ	○
		mediaType	1	クラスタ	○
		linkSpeed	1	クラスタ	○
	PrivateMaintenanceLans		1	クラスタ	○

プロパティ			出力される個数	サポート構成	verbose が false の場合の表示	
	--	PrivateMaintenanceLan		0, 1 ～ n	クラスタ	○
		--	status	1	クラスタ	○
			portName	1	クラスタ	○
			mediaType	1	クラスタ	○
			linkSpeed	1	クラスタ	○
	InternalHDDs			1	クラスタ	○
	--	InternalHDD		0, 1 ～ n	クラスタ	○
		--	slotName	1	クラスタ	○
			status	1	クラスタ	○
			vendorInformation	1	クラスタ	○
	InternalRAIDBatteries			1	クラスタ	○
	--	InternalRAIDBattery		0, 1 ～ n	クラスタ	○
		--	slotName	1	クラスタ	○
			status	1	クラスタ	○
			vendorInformation	1	クラスタ	○
	Memories			1	クラスタ	○
	--	Memory		0, 1 ～ n	クラスタ	○
		--	slotName	1	クラスタ	○
			status	1	クラスタ	○
			vendorInformation	1	クラスタ	○
	MemTotals			1	クラスタ	○
	--	MemTotal		0, 1	クラスタ	○
		--	slotName	1	クラスタ	○
			size	1	クラスタ	○
	PowerSupplies			1	クラスタ	○
	--	PowerSupply		0, 1 ～ n	クラスタ	○
		--	powerSupplyName	1	クラスタ	○
			status	1	クラスタ	○
			vendorInformation	1	クラスタ	○
	Fans			1	クラスタ	○
	--	Fan		0, 1 ～ n	クラスタ	○
		--	fanName	1	クラスタ	○
			status	1	クラスタ	○
			vendorInformation	1	クラスタ	○
	Temperatures			1	クラスタ	○
	--	Temperature		0, 1 ～ n	クラスタ	○
		--	temperatureName	1	クラスタ	○
			status	1	クラスタ	○
			vendorInformation	1	クラスタ	○

プロパティ				出力される個数	サポート構成	verbose が false の場合の表示
	BMCs			1	クラスタ	○
	--	BMC		0, 1	クラスタ	○
		--	status	1	クラスタ	○
			connection	0, 1	クラスタ	○
	Models			1	クラスタ	○
	--	Model		0, 1	クラスタ	○
		--	manufacturer	1	クラスタ	○
			product	1	クラスタ	○
	Serials			1	クラスタ	○
	--	Serial		0, 1	クラスタ	○
		--	number	1	クラスタ	○

(凡例) n : 2 以上の整数を示します。○ : 表示されます。

Hardware リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

表 3-49 : Hardware リソースに GET メソッドをリクエストしたときのプロパティ

プロパティ				説明
Hardware				--
--	PortDetails			--
	--	PortDetail		--
		--	status	データポートの状態が表示されます。 Up リンクが正常な場合に表示されます。 Down リンクが断絶している場合に表示されます。
			portName	データポートの名称が次のどちらかの形式で表示されます。 eth <番号> または xgbe <番号>
			mediaType	データポートのメディアタイプが表示されます。 copper メタルケーブルに対応している場合に表示されます。 fiber 光ケーブルに対応している場合に表示されます。
			linkSpeed	データポートの通信速度が表示されます。 10Base 10Mbps の場合に表示されます。 100Base 100Mbps の場合に表示されます。 1000Base 1Gbps の場合に表示されます。 10000Base 10Gbps の場合に表示されます。 Unknown! 通信速度が不明の場合に表示されます。

プロパティ				説明
	ManagementLans			--
	--	ManagementLan		--
		--	status	管理ポートの状態が表示されます。 Up リンクが正常な場合に表示されます。 Down リンクが断絶している場合に表示されます。
			portName	管理ポートの名称（mng0）が表示されます。
			mediaType	管理ポートのメディアタイプが表示されます。 copper メタルケーブルに対応している場合に表示されます。 fiber 光ケーブルに対応している場合に表示されます。
			linkSpeed	管理ポートの通信速度が表示されます。 10Base 10Mbps の場合に表示されます。 100Base 100Mbps の場合に表示されます。 1000Base 1Gbps の場合に表示されます。 10000Base 10Gbps の場合に表示されます。 Unknown! 通信速度が不明の場合に表示されます。
	HeartbeatLans			クラスタ構成の場合だけ、表示されます。
	--	HeartbeatLan		--
		--	status	ハートビートポートの状態が表示されます。 Up リンクが正常な場合に表示されます。 Down リンクが断絶している場合に表示されます。
			portName	ハートビートポートの名称（hb0）が表示されます。
			mediaType	ハートビートポートのメディアタイプが表示されます。 copper メタルケーブルに対応している場合に表示されます。 fiber 光ケーブルに対応している場合に表示されます。



プロパティ				説明
			linkSpeed	ハートビートポートの通信速度が表示されます。 10Base 10Mbps の場合に表示されます。 100Base 100Mbps の場合に表示されます。 1000Base 1Gbps の場合に表示されます。 10000Base 10Gbps の場合に表示されます。 Unknown! 通信速度が不明の場合に表示されます。
	PrivateMaintenanceLans			--
	--	PrivateMaintenanceLan		--
		--	status	保守ポートの状態が表示されます。 Up リンクが正常な場合に表示されます。 Down リンクが断絶している場合に表示されます。
			portName	保守ポートの名称が次の形式で表示されます。 <b>pm</b> <番号>
			mediaType	保守ポートのメディアタイプが表示されます。 copper メタルケーブルに対応している場合に表示されます。 fiber 光ケーブルに対応している場合に表示されます。
			linkSpeed	保守ポートの通信速度が表示されます。 10Base 10Mbps の場合に表示されます。 100Base 100Mbps の場合に表示されます。 1000Base 1Gbps の場合に表示されます。 10000Base 10Gbps の場合に表示されます。 Unknown! 通信速度が不明の場合に表示されます。
	InternalHDDs			--
	--	InternalHDD		--
		--	slotName	内蔵ハードディスクのインデックス番号が 0 以上の整数で表示されます。
			status	内蔵ハードディスクの状態が表示されます。 Normal 正常な場合に表示されます。 Rebuild RAID の再構築中の場合に表示されます。 Failed 障害が発生している場合に表示されます。 Not supported 情報を取得するためのプログラムがインストールされていない場合に表示されます。

プロパティ				説明
			vendorInformation	内蔵ハードディスクのベンダー固有の詳細情報が表示されます。
	InternalRAIDBatteries			--
	--	InternalRAIDBattery		--
		--	slotName	内蔵 RAID バッテリーのインデックス番号が 0 以上の整数で表示されます。
			status	内蔵 RAID バッテリーの状態が表示されます。 OK 充電済みの場合に表示されます。 Charging 充電中の場合に表示されます。 Failed 障害が発生している場合に表示されます。 Not supported 情報を取得するためのプログラムがインストールされていない場合に表示されます。
			vendorInformation	内蔵 RAID バッテリーのベンダー固有の詳細情報が表示されます。
	Memories			--
	--	Memory		--
		--	slotName	メモリーのインデックス番号が 0 以上の整数で表示されます。
			status	メモリーの状態が表示されます。 Installed メモリーが搭載されている場合に表示されます。 Not Installed メモリーが搭載されていない場合に表示されます。
			vendorInformation	メモリーのベンダー固有の詳細情報が表示されます。
	MemTotals			--
	--	MemTotal		--
		--	slotName	メモリーの合計容量のインデックス番号が 0 以上の整数で表示されます。
			size	メモリーの合計容量が表示されます。
	PowerSupplies			--
	--	PowerSupply		--
		--	powerSupplyName	電源ユニットのインデックス番号が 0 以上の整数で表示されます。
			status	電源ユニットの状態が表示されます。 OK 正常な場合に表示されます。 Failed 障害が発生している場合に表示されます。 Not Installed 電源ユニットが搭載されていない場合に表示されます。
			vendorInformation	電源ユニットのベンダー固有の詳細情報が表示されます。

プロパティ			説明	
	Fans		--	
	--	Fan		--
		--	fanName	ファンのインデックス番号が 0 以上の整数で表示されます。
			status	ファンの状態が表示されます。 OK 正常な場合に表示されます。 Failed 障害が発生している場合またはファンが搭載されていない場合に表示されます。
			vendorInformation	ファンのベンダー固有の詳細情報が表示されます。
	Temperatures		--	
	--	Temperature		--
		--	temperatureName	温度センサーのインデックス番号が 0 以上の整数で表示されます。
			status	温度センサーの検知結果が表示されます。 OK 正常な場合に表示されます。 Failed 異常な場合に表示されます。
			vendorInformation	温度センサーのベンダー固有の詳細情報が表示されます。
	BMCs		--	
	--	BMC		--
		--	status	リクエスト先のノードの BMC の状態が表示されます。 OK 正常な場合に表示されます。 Unknown! 情報を取得できなかった場合に表示されます。
			connection	クラスタ構成の場合は、リクエスト先のノードの相手ノードとの接続の状態が表示されます。 OK 正常な場合に表示されます。 Failed 相手ノードの BMC との通信に失敗した場合に表示されます。 None クラスタが構築されていない場合または情報取得に失敗した場合に表示されます。
		Models		--
	--	Model		--
		--	manufacturer	製造元が表示されます。
			product	製品名が表示されます。
	Serials		--	
	--	Serial		--
--		number	装置識別番号が 64 バイト以内で表示されます。	

注 装置が搭載されていない場合、対象のプロパティは表示されません。また、情報を取得できない場合、対象のプロパティに空文字列が表示されます。

Hardware リソースに GET メソッドをリクエストして、ハードウェアのクラスタモードの詳細を取得する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/Hardware

### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Hardware>
  <PortDetails>
    <PortDetail>
      <status>Up</status>
      <portName>eth1</portName>
      <mediaType>copper</mediaType>
      <linkSpeed>1000Base</linkSpeed>
    </PortDetail>
    <PortDetail>
      <status>Down</status>
      <portName>eth2</portName>
      <mediaType>copper</mediaType>
      <linkSpeed>Unknown!</linkSpeed>
    </PortDetail>
  </PortDetails>
  <ManagementLans>
    <ManagementLan>
      <status>Up</status>
      <portName>mng0</portName>
      <mediaType>copper</mediaType>
      <linkSpeed>100Base</linkSpeed>
    </ManagementLan>
  </ManagementLans>
  <HeartbeatLans>
    <HeartbeatLan>
      <status>Up</status>
      <portName>hb0</portName>
      <mediaType>copper</mediaType>
      <linkSpeed>100Base</linkSpeed>
    </HeartbeatLan>
  </HeartbeatLans>
  <PrivateMaintenanceLans>
    <PrivateMaintenanceLan>
      <status>Up</status>
      <portName>pm0</portName>
      <mediaType>copper</mediaType>
      <linkSpeed>100Base</linkSpeed>
    </PrivateMaintenanceLan>
    <PrivateMaintenanceLan>
      <status>Down</status>
      <portName>pm1</portName>
      <mediaType>copper</mediaType>
      <linkSpeed>Unknown!</linkSpeed>
    </PrivateMaintenanceLan>
  </PrivateMaintenanceLans>
  <InternalHDDs>
    <InternalHDD>
      <slotName>0</slotName>
      <status>Normal</status>
      <vendorInformation>size:136.732 GB,raid level:RAID1</
vendorInformation>
    </InternalHDD>
    <InternalHDD>
      <slotName>1</slotName>
      <status>Normal</status>
      <vendorInformation>size:136.732 GB,raid level:RAID1</
vendorInformation>
    </InternalHDD>
  </InternalHDDs>
  <InternalRAIDBatteries/>
  <Memories>
    <Memory>
      <slotName>0</slotName>
      <status>Not Installed</status>
      <vendorInformation>locator:P1C0D1,size:-</vendorInformation>
    </Memory>
  </Memories>
</Hardware>
```

```

        </Memory>
        <Memory>
            <slotName>1</slotName>
            <status>Installed</status>
            <vendorInformation>locator:P1C0D0,size:2048 MB</
vendorInformation>
        </Memory>
    </Memories>
    <MemTotals>
        <MemTotal>
            <slotName>0</slotName>
            <size>size:11.5 GB(12.0 GB)</size>
        </MemTotal>
    </MemTotals>
    <PowerSupplies>
        <PowerSupply>
            <powerSupplyName>0</powerSupplyName>
            <status>OK</status>
            <vendorInformation></vendorInformation>
        </PowerSupply>
    </PowerSupplies>
    <Fans>
        <Fan>
            <fanName>0</fanName>
            <status>OK</status>
            <vendorInformation></vendorInformation>
        </Fan>
        <Fan>
            <fanName>1</fanName>
            <status>OK</status>
            <vendorInformation></vendorInformation>
        </Fan>
    </Fans>
    <Temperatures>
        <Temperature>
            <temperatureName>0</temperatureName>
            <status>OK</status>
            <vendorInformation>reading:55 (+/- 0.500) degrees C</
vendorInformation>
        </Temperature>
        <Temperature>
            <temperatureName>1</temperatureName>
            <status>OK</status>
            <vendorInformation>reading:44 (+/- 0.500) degrees C</
vendorInformation>
        </Temperature>
    </Temperatures>
    <BMCs>
        <BMC>
            <status>OK</status>
            <connection>OK</connection>
        </BMC>
    </BMCs>
    <Models>
        <Model>
            <manufacturer>NEC</manufacturer>
            <product>Express5800/R120d-2M [N8100-1963Y]</product>
        </Model>
    </Models>
    <Serials>
        <Serial>
            <number>4230MB0-T019000037</number>
        </Serial>
    </Serials>
</Hardware>

```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-50 : Hardware リソースに GET メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

## 3.7 LU

### 3.7.1 LU リソースの概要

LU リソースを使用して、LU の情報を取得できます。

LU リソースに使用できる HTTP メソッドを次の表に示します。

表 3-51：LU リソースに使用できる HTTP メソッド

リソースの URI	HTTP メソッド	サポート構成	説明
/LUs	GET	クラスタ Virtual Server	割り当てられているすべての LU の情報を取得します。 Virtual Server 上で実行した場合、ログインしている Virtual Server に割り当てられている LU の情報だけを取得します。

### 3.7.2 LU リソース

この項では、LU リソースの使い方について説明します。

#### 3.7.2.1 GET メソッドで LU の情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

表 3-52：LU リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ		出力される個数	サポート構成	verbose が false の場合の表示
LUs		1	クラスタ Virtual Server	○
--	LU	0, 1 ~ n	クラスタ Virtual Server	○
	--			
	deviceFileName	1	クラスタ Virtual Server	○
	userLUN	1	クラスタ Virtual Server	○
	isExternalDevice	1	クラスタ Virtual Server	○
	target	1	クラスタ Virtual Server	○
	model	1	クラスタ Virtual Server	○
	serial	1	クラスタ Virtual Server	○
	ldev	1	クラスタ Virtual Server	○
	type	1	クラスタ Virtual Server	○
	size	1	クラスタ Virtual Server	○
	function	1	クラスタ Virtual Server	○

プロパティ			出力される個数	サポート構成	verbose が false の場合の表示
		usedBy	1	クラスタ Virtual Server	○
		rg	1	クラスタ Virtual Server	○
		dp	1	クラスタ Virtual Server	○
		usedPage	1	クラスタ Virtual Server	○
		pool	1	クラスタ Virtual Server	○
		freePool	1	クラスタ Virtual Server	○
		totalPool	1	クラスタ Virtual Server	○

(凡例) n : 2 以上の整数を示します。○ : 表示されます。

LU リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

**表 3-53 : LU リソースに GET メソッドをリクエストしたときのプロパティ**

プロパティ			説明
LUs			--
--	LU		--
	--	deviceFileName	デバイスファイルの名称が次の形式で表示されます。 lu <ユーザー LUN >
		userLUN	ユーザー LUN が 16 進数で表示されます。 デバイスファイルが外部ストレージの場合は、数値の後ろに番号記号 (#) が表示されます。
		isExternalDevice	デバイスファイルが外部ストレージかどうかが表示されます。 true 外部ストレージの場合に表示されます。 false 外部ストレージでない場合に表示されます。
		target	ターゲットが次の形式で表示されます。 N <ノード番号> -T < 000 ~ 255 >
		model	ストレージシステムのモデルが表示されます。
		serial	ストレージシステムのシリアル番号が 16 バイト以内で表示されます。
		ldev	デバイスファイルの LDEV 番号が 5 バイト以内で表示されます。 ユーザー LUN に LDEV 番号が割り当てられていない場合、または内蔵ハードディスクの場合は、空文字列が表示されます。

プロパティ			説明
		type	<p>デバイスファイルのドライブ種別が表示されます。</p> <p>FC/SAS</p> <p>FC または SAS ドライブの場合に表示されます。</p> <p>SAS7K</p> <p>SAS 7.2K ドライブの場合に表示されます。</p> <p>SATA</p> <p>SATA ドライブの場合に表示されます。</p> <p>SSD</p> <p>SSD の場合に表示されます。</p> <p>次の場合は、空文字列が表示されます。</p> <ul style="list-style-type: none"> <li>・ 障害のため情報を取得できない場合</li> <li>・ 階層プール上の LU を使用している場合</li> </ul>
		size	<p>デバイスファイルのサイズが GB 単位で表示されます。</p> <p>デバイスファイルに障害が発生している場合は、ハイフン (-) が表示されます。</p> <p>小数点第 4 位以下は切り捨てて表示されます。</p>
		function	<p>デバイスファイルの用途が表示されます。</p> <p>Free</p> <p>未使用のデバイスファイルの場合に表示されます。</p> <p>File</p> <p>ファイルシステムで使用しているデバイスファイルの場合に表示されます。</p> <p>Snap, P-VOL</p> <p>ファイルスナップショット機能で運用されているファイルシステムで使用しているデバイスファイルの場合に表示されます。</p> <p>Snap, D-VOL</p> <p>差分格納デバイスで使用しているデバイスファイルの場合に表示されます。</p> <p>HFRR, P-VOL</p> <p>File Remote Replicator で運用されているファイルシステムで使用しているデバイスファイルの場合に表示されます。</p> <p>HFRR, D-VOL</p> <p>File Remote Replicator で運用されている差分格納デバイスで使用しているデバイスファイルの場合に表示されます。</p> <p>V_OSLU</p> <p>Virtual Server OS LU の場合に表示されます。</p> <p>V_UserLU</p> <p>Virtual Server ユーザー LU の場合に表示されます。</p> <p>WORM</p> <p>WORM 対応ファイルシステムで使用しているデバイスファイルの場合に表示されます。</p> <p>WORM, Snap, P-VOL</p> <p>ファイルスナップショット機能で運用されている WORM 対応ファイルシステムで使用しているデバイスファイルの場合に表示されます。</p> <p>WORM, Snap, D-VOL</p> <p>ファイルスナップショット機能で運用されている WORM 対応ファイルシステムの差分格納デバイスで使用しているデバイスファイルの場合に表示されます。</p>



プロパティ			説明
			<p>WORM, HFRR, P-VOL</p> <p>File Remote Replicator で運用されている WORM 対応ファイルシステムで使用しているデバイスファイルの場合に表示されます。</p> <p>WORM, HFRR, D-VOL</p> <p>File Remote Replicator で運用されている WORM 対応ファイルシステムの差分格納デバイスで使用しているデバイスファイルの場合に表示されます。</p> <p>Defined</p> <p>Backup Restore のボリュームレプリケーション連携機能で、ファイルシステムのコピー先として予約されているデバイスファイルの場合に表示されます。</p> <p>Unknown</p> <p>LU の状況が次のどちらかの場合に表示されます。</p> <ul style="list-style-type: none"> <li>- コマンド実行中にデバイスファイルの用途が変わった</li> <li>- 障害が発生してデバイスファイルの情報が取得できなかった</li> </ul> <p>Unknown が表示された場合は、しばらくしてから再度実行してください。それでも Unknown が表示される場合は、システムメッセージを確認して障害を取り除いてください。システムメッセージが出力されていない場合は、カスタマーサポートセンターに連絡してください。</p> <p>また、Virtual Server に割り当てたユーザー LU の容量を変更した場合や、Virtual Server へのユーザー LU の割り当てが失敗した場合に、Unknown が表示されることがあります。この場合、Virtual Server へのデバイスファイルの割り当てをいったん解除したあと、Processing Node のリフレッシュ処理を実行し、再度 Virtual Server にデバイスファイルを割り当ててください。</p>
		usedBy	デバイスファイルの使用リソースの名称が表示されます。function プロパティが File, WORM, Snap, HFRR の場合にファイルシステム名が表示されます。V_OSLU, V_UserLU の場合は Virtual Server 名が表示されます。
		rg	—
		dp	—
		usedPage	—
		pool	—
		freePool	—
		totalPool	—

LU リソースに GET メソッドをリクエストして、LU の情報を取得する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/LUs

#### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<LUs>
  <LU>
    <deviceFileName>lu00</deviceFileName>
    <userLUN>00</userLUN>
    <isExternalDevice>false</isExternalDevice>
    <target>N1-T000</target>
    <serial>83016761</serial>
    <ldev>1</ldev>
    <type>SATA</type>
  </LU>
</LUs>
```

```

<size>20.000</size>
<function>File</function>
<usedBy>MyShare</usedBy>
<rg>000</rg>
<dp></dp>
<usedPage></usedPage>
<pool></pool>
<freePool></freePool>
<totalPool></totalPool>
</LU>
<LU>
  <deviceFileName>lu01</deviceFileName>
  <userLUN>01</userLUN>
  <isExternalDevice>false</isExternalDevice>
  <target>N1-T000</target>
  <serial>83016761</serial>
  <ldev>2</ldev>
  <type>SATA</type>
  <size>20.000</size>
  <function>File</function>
  <usedBy>test</usedBy>
  <rg>000</rg>
  <dp></dp>
  <usedPage></usedPage>
  <pool></pool>
  <freePool></freePool>
  <totalPool></totalPool>
</LU>
</LUs>

```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-54： LU リソースに GET メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3：」を参照してください。

## 3.8 NetworkInterface

### 3.8.1 NetworkInterface リソースの概要

NetworkInterface リソースを使用して、ネットワークインターフェースの情報を取得したり、管理したりできます。

NetworkInterface リソースに使用できる HTTP メソッドを次の表に示します。

表 3-55： NetworkInterface リソースに使用できる HTTP メソッド

リソースの URI	HTTP メソッド	サポート構成	説明
/NetworkInterfaces	GET	クラスタ Virtual Server	現在設定されているネットワークインターフェースの情報を取得します。 Virtual Server 上で実行した場合は、ログインしている VirtualServer に関するインターフェースの情報だけを取得します。
	PUT	クラスタ	ネットワークインターフェースを追加します。

リソースの URI	HTTP メソッド	サポート構成	説明
/NetworkInterfaces/ <ネットワークインター フェース名>	GET	クラスタ Virtual Server	指定したネットワークインター フェースの情報を取得します。
	POST	クラスタ	指定したネットワークインター フェースの情報を編集します。
	HEAD	クラスタ	指定したネットワークインター フェースが存在するかどうかを確認 します。

## 3.8.2 NetworkInterface リソース

この項では、NetworkInterface リソースの使い方について説明します。

### 3.8.2.1 GET メソッドでネットワークインターフェースの情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

**表 3-56 : NetworkInterface リソースに GET メソッドをリクエストしたときのレスポンス XML の構造**

プロパティ			出力される個数	サポート構成	verbose が false の場合の表示
NetworkInterfaces			1	クラスタ Virtual Server	○
--	NetworkInterface		0, 1 ～ n	クラスタ Virtual Server	○
	--	name	1	クラスタ Virtual Server	○
		mtu	1	クラスタ Virtual Server	×
		netmask	1	クラスタ Virtual Server	○
		networkAddress	1	クラスタ Virtual Server	×
		prefixLength	1	クラスタ Virtual Server	○
		addressPrefix	1	クラスタ Virtual Server	×
		FixedIP		1	クラスタ
	--	target	1	クラスタ	○
		other	1	クラスタ	○
		targetIPv6	1	クラスタ	○
		otherIPv6	1	クラスタ	○
	ServiceIP		1	クラスタ Virtual Server	○
	--	target	1	クラスタ Virtual Server	○
		other	1	クラスタ	○
		targetIPv6	1	クラスタ Virtual Server	○
		otherIPv6	1	クラスタ	○

(凡例) n : 2 以上の整数を示します。○ : 表示されます。× : 表示されません。

NetworkInterface リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

**表 3-57 : NetworkInterface リソースに GET メソッドをリクエストしたときのプロパティ**

プロパティ			説明
NetworkInterfaces			--
--	NetworkInterface		--
	--	name	インターフェースの名称が表示されます。
		mtu <sup>*1</sup>	MTU 値が 5 バイト以内で表示されます。 インターフェースが設定されていない場合は、空文字列が表示されます。
		netmask <sup>*2</sup>	ネットマスクが表示されます。 インターフェースが設定されていない場合は、空文字列が表示されます。
		networkAddress	ネットワークアドレスが表示されます。 インターフェースが設定されていない場合は、空文字列が表示されます。
		prefixLength	IPv6 のプレフィックス長が表示されます。 インターフェースが設定されていない場合は、空文字列が表示されます。
		addressPrefix	IPv6 のアドレスプレフィックスが表示されます。 インターフェースが設定されていない場合は、空文字列が表示されます。
		FixedIP	--
	--	target	リクエスト先のノードの固有 IP アドレスが表示されます。 未使用 LAN ポートの場合は、空文字列が表示されます。
		other <sup>*2</sup>	リクエスト先のノードの相手ノードの固有 IP アドレスが表示されます。 未使用 LAN ポートの場合は、空文字列が表示されます。
		targetIPv6	リクエスト先のノードの固有 IP アドレスが表示されます。 未使用 LAN ポートの場合は、空文字列が表示されます。
		otherIPv6	リクエスト先のノードの相手ノードの固有 IP アドレスが表示されます。 未使用 LAN ポートの場合は、空文字列が表示されます。
	ServiceIP		--
	--	target	リクエスト先のノードの仮想 IP アドレスが表示されます。 仮想 IP アドレスが設定されていない場合は、空文字列が表示されます。
		other <sup>*2</sup>	リクエスト先のノードの相手ノードの仮想 IP アドレスが表示されます。 仮想 IP アドレスが設定されていない場合は、空文字列が表示されます。
		targetIPv6	リクエスト先のノードの仮想 IP アドレスが表示されます。 仮想 IP アドレスが設定されていない場合は、空文字列が表示されます。
		otherIPv6	リクエスト先のノードの相手ノードの仮想 IP アドレスが表示されます。 仮想 IP アドレスが設定されていない場合は、空文字列が表示されます。

注 \*1 クラスタ内で MTU 値が異なる場合は Invalid( <メソッドの実行対象のノードの MTU 値> , <もう一方のノードの MTU 値> ) と表示されます。

ノード上の OS が起動していない、通信でエラーが発生しているなどの理由で MTU 値を取得できない場合は、Unknown と表示されます。

例：Invalid(1500,Unknown)

そのほかの理由で MTU 値を取得できない場合は、None と表示されます。

例：Invalid(1500,None)

注 \*2 OS が起動していないまたは通信でエラーが発生しているなどの理由で情報を取得できない場合は、Unknown と表示されます。また、一方のノードだけにインターフェースを設定している、同一インターフェース内でネットワークアドレスが異なる IP アドレスが設定されているなどの理由で、取得した情報に誤りがある場合は Invalid( <不正値> ) と表示されます。

NetworkInterface リソースに GET メソッドをリクエストして、ネットワークインターフェースの情報を取得する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/NetworkInterfaces

### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NetworkInterface>
  <name>mng0</name>
  <mtu>1500</mtu>
  <netmask>255.255.0.0</netmask>
  <networkAddress>192.168.0.0</networkAddress>
  <prefixLength>64</prefixLength>
  <addressPrefix>fdf1:afcb:44f9:1000::/64</addressPrefix>
  <dhcp>Off</dhcp>
  <FixedIP>
    <target>192.168.2.10</target>
    <other>192.168.2.20</other>
    <targetIPv6>fdf1:afcb:44f9:1000::2:0</targetIPv6>
    <otherIPv6>fdf1:afcb:44f9:1000::2:1</otherIPv6>
  </FixedIP>
  <ServiceIP>
    <target>192.168.2.11</target>
    <other>192.168.2.21</other>
    <targetIPv6>fdf1:afcb:44f9:1000::2:2</targetIPv6>
    <otherIPv6>fdf1:afcb:44f9:1000::2:3</otherIPv6>
  </ServiceIP>
</NetworkInterface>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-58 : NetworkInterface リソースに GET メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.8.2.2 PUT メソッドでネットワークインターフェースを追加する

NetworkInterface リソースに PUT メソッドをリクエストするときの XML の構造を次の表に示します。

表 3-59 : NetworkInterface リソースに PUT メソッドをリクエストするときの XML の構造

プロパティ		データ型	指定できる個数	サポート構成	
NetworkInterface		--	1	クラスタ	
--	name	String	1	クラスタ	
	netmask	String	0, 1	クラスタ	
	prefixLength	String	0, 1	クラスタ	
	FixedIP	--	1	クラスタ	
	--	target	String	0, 1	クラスタ
		other	String	0, 1	クラスタ
		targetIPv6	String	0, 1	クラスタ
		otherIPv6	String	0, 1	クラスタ
	ServiceIP	--	0, 1	クラスタ	
	--	target	String	0, 1	クラスタ
		other	String	0, 1	クラスタ
		targetIPv6	String	0, 1	クラスタ
		otherIPv6	String	0, 1	クラスタ

NetworkInterface リソースに PUT メソッドをリクエストするときに指定するプロパティについて次の表に示します。

表 3-60 : NetworkInterface リソースに PUT メソッドをリクエストするときのプロパティ

プロパティ		説明	指定時の条件	
NetworkInterface		--	--	
--	name	ネットワークインターフェースを追加するポートの名称を設定します。GET メソッドの実行結果に表示されているネットワークインターフェースのうち、name プロパティのほかに設定がないネットワークインターフェースだけ、設定できます。 <ul style="list-style-type: none"><li>• eth &lt;番号&gt;</li><li>• agr &lt;番号&gt;</li><li>• rdn &lt;番号&gt;</li><li>• xgbe &lt;番号&gt;</li></ul>	必須	
	netmask	固有 IP アドレスおよび仮想 IP アドレスのネットマスクを設定します。	IPv4 を使用する場合は必須です。	
	prefixLength	IPv6 のプレフィックス長を設定します。	IPv6 を使用する場合は必須です。	
	FixedIP*1		--	必須
	--	target	リクエスト先のノードの固有 IP アドレスを設定します。	IPv4 形式のアドレスを使用する場合は必須です。
		other	リクエスト先のノードの相手ノードの固有 IP アドレスを設定します。	IPv4 形式のアドレスを使用する場合は必須です。
targetIPv6		リクエスト先のノードの固有 IP アドレスを IPv6 で設定します。	IPv6 形式のアドレスを使用する場合は必須です。	

プロパティ			説明	指定時の条件
		otherIPv6	リクエスト先のノードの相手ノードの固有 IP アドレスを IPv6 で設定します。	IPv6 形式のアドレスを使用する場合は必須です。
	ServiceIp*1*2		--	任意
	--	target	リクエスト先のノードの仮想 IP アドレスを設定します。空文字列を設定した場合、および省略した場合は値が設定されません。	IPv4 形式のアドレスを使用する場合は必須です。
		other	リクエスト先のノードの相手ノードの仮想 IP アドレスを設定します。空文字列を設定した場合、および省略した場合は値が設定されません。	IPv4 形式のアドレスを使用する場合は必須です。
		targetIPv6	リクエスト先のノードの仮想 IP アドレスを IPv6 で設定します。空文字列を設定した場合、および省略した場合は値が設定されません。	固有 IP アドレスに IPv6 形式のアドレスを使用する場合は必須です。
		otherIPv6	リクエスト先のノードの相手ノードの仮想 IP アドレスを IPv6 で設定します。空文字列を設定した場合、および省略した場合は値が設定されません。	固有 IP アドレスに IPv6 形式のアドレスを使用する場合は必須です。

注 \*1 次のネットワークの IP アドレスは指定しないでください。これらの IP アドレスはストレージシステム内部で予約されています。

- IPv4 の場合 :  
127.0.0.0 ~ 127.255.255.255
- IPv6 の場合 :  
::ffff: < IPv4 アドレス >, :: < IPv4 アドレス >, ::1/128, ::/0, ::/ 128, fe80::/10, ff00::/8
- 保守ポートに設定されている IP アドレスのネットワーク  
保守ポートに設定されている IP アドレスは pmctl コマンドで確認できます。
- ハートビートポートに設定されている IP アドレスのネットワーク  
ハートビートポートに設定されている IP アドレスは、ProcessingNode リソースを使用して確認してください。ProcessingNode リソースについては、「[3.10.2 ProcessingNode リソース](#)」を参照してください。

注 \*2 仮想 IP アドレスを指定しないで作成したインターフェースは、リソースグループの監視の対象外となります。仮想 IP アドレスを指定しないでインターフェースを作成するときの注意事項を次に示します。

- 仮想 IP アドレスが設定されていないインターフェースから、HVFP が提供する CIFS サービス、NFS サービス、FTP サービス、SFTP サービスおよび TFTP サービスを利用できません。
- 仮想 IP アドレスが設定されていないインターフェースでリンク障害が発生すると、SNMP トラップまたは E-mail 通知が出力されません。

NetworkInterface リソースに PUT メソッドをリクエストして、ネットワークインターフェースを追加する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/NetworkInterfaces

#### リクエスト XML の例 (IPv4 の場合)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NetworkInterface>
  <name>eth1</name>
  <netmask>255.255.255.0</netmask>
  <FixedIP>
    <target>10.208.136.210</target>
    <other>10.208.136.211</other>
  </FixedIP>
  <ServiceIP>
    <target>10.208.136.212</target>
    <other>10.208.136.213</other>
  </ServiceIP>
</NetworkInterface>
```

#### リクエスト XML の例 (IPv6 の場合)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NetworkInterface>
  <name>eth1</name>
  <prefixLength>64</prefixLength>
  <FixedIP>
    <targetIPv6>2001:2c0:418:1::1</targetIPv6>
    <otherIPv6>2001:2c0:418:1::2</otherIPv6>
  </FixedIP>
  <ServiceIP>
    <targetIPv6>2001:2c0:418:1::3</targetIPv6>
    <otherIPv6>2001:2c0:418:1::4</otherIPv6>
  </ServiceIP>
</NetworkInterface>
```

#### リクエスト XML の例 (固有 IP アドレスを設定しない場合)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NetworkInterface>
  <name>eth1</name>
  <netmask>255.255.255.0</netmask>
  <FixedIP>
    <target></target>
    <other></other>
  </FixedIP>
  <ServiceIP>
    <target></target>
    <other></other>
  </ServiceIP>
</NetworkInterface>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-61 : NetworkInterface リソースに PUT メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

## 3.8.3 NetworkInterfaces/ <ネットワークインターフェース名> リソース

この項では、NetworkInterfaces/ <ネットワークインターフェース名> リソースの使い方について説明します。

### 3.8.3.1 GET メソッドでネットワークインターフェースの情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。



表 3-62 : NetworkInterfaces/ <ネットワークインターフェース名>リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ		出力される個数	サポート構成	verbose が false の場合の表示
NetworkInterface		1	クラスタ Virtual Server	○
--	name	1	クラスタ Virtual Server	○
	mtu	1	クラスタ Virtual Server	×
	netmask	1	クラスタ Virtual Server	○
	networkAddress	1	クラスタ Virtual Server	×
	prefixLength	1	クラスタ Virtual Server	○
	addressPrefix	1	クラスタ Virtual Server	×
	FixedIP	1	クラスタ	○
	--	target	クラスタ	○
		other	クラスタ	○
		targetIPv6	クラスタ	○
		otherIPv6	クラスタ	○
	ServiceIP	1	クラスタ Virtual Server	○
	--	target	クラスタ Virtual Server	○
		other	クラスタ	○
		targetIPv6	クラスタ Virtual Server	○
		otherIPv6	クラスタ	○

(凡例) ○ : 表示されます。× : 表示されません。

NetworkInterfaces/ <ネットワークインターフェース名>リソースに GET メソッドをリクエストしたときに表示されるプロパティについては、「表 3-57 : NetworkInterface リソースに GET メソッドをリクエストしたときのプロパティ」を参照してください。

NetworkInterfaces/ <ネットワークインターフェース名>リソースに GET メソッドをリクエストして、指定したネットワークインターフェースの情報を取得する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/NetworkInterfaces/<ネットワークインターフェース名>

#### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NetworkInterface>
  <name>mng0</name>
  <mtu>1500</mtu>
  <netmask>255.255.0.0</netmask>
  <networkAddress>192.168.0.0</networkAddress>
  <prefixLength>64</prefixLength>
  <addressPrefix>fd1:afcb:44f9:1000::/64</addressPrefix>
  <dhcp>Off</dhcp>
  <FixedIP>
    <target>192.168.2.10</target>
```

```

        <other>192.168.2.20</other>
        <targetIPv6>fdf1:afcb:44f9:1000::2:0</targetIPv6>
        <otherIPv6>fdf1:afcb:44f9:1000::2:1</otherIPv6>
    </FixedIP>
    <ServiceIP>
        <target>192.168.2.11</target>
        <other>192.168.2.21</other>
        <targetIPv6>fdf1:afcb:44f9:1000::2:2</targetIPv6>
        <otherIPv6>fdf1:afcb:44f9:1000::2:3</otherIPv6>
    </ServiceIP>
</NetworkInterface>

```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-63 : NetworkInterfaces/ <ネットワークインターフェース名>リソースに GET メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.8.3.2 POST メソッドでネットワークインターフェースの情報を変更する

NetworkInterfaces/ <ネットワークインターフェース名>リソースに POST メソッドをリクエストするときの XML の構造を次の表に示します。

**表 3-64 : NetworkInterfaces/ <ネットワークインターフェース名>リソースに POST メソッドをリクエストするときの XML の構造**

プロパティ		データ型	指定できる個数	サポート構成
NetworkInterface		--	1	クラスタ
--	netmask	String	0, 1	クラスタ
	prefixLength	String	0, 1	クラスタ
	FixedIP	--	0, 1	クラスタ
	--	target	0, 1	クラスタ
		other	0, 1	クラスタ
		targetIPv6	0, 1	クラスタ
		otherIPv6	0, 1	クラスタ
	ServiceIP	--	0, 1	クラスタ
	--	target	0, 1	クラスタ
		other	0, 1	クラスタ
		targetIPv6	0, 1	クラスタ
		otherIPv6	0, 1	クラスタ

NetworkInterfaces/ <ネットワークインターフェース名>リソースに POST メソッドをリクエストするとき指定するプロパティについて次の表に示します。

**表 3-65 : NetworkInterfaces/ <ネットワークインターフェース名>リソースに POST メソッドをリクエストするときのプロパティ**

プロパティ	説明	指定時の条件
NetworkInterface	--	--

プロパティ		説明	指定時の条件
--	netmask	固有 IP アドレスと仮想 IP アドレスのネットマスクを設定します。	次のプロパティのうち、どれか一つを必ず指定します。 <ul style="list-style-type: none"> <li>• netmask</li> <li>• prefixLength</li> <li>• FixedIP</li> <li>• ServiceIP</li> </ul>
	prefixLength	IPv6 のプレフィックス長を設定します。	IPv6 形式のアドレスを使用する場合は必須です。 次のプロパティのうち、どれか一つを必ず指定します。 <ul style="list-style-type: none"> <li>• netmask</li> <li>• prefixLength</li> <li>• FixedIP</li> <li>• ServiceIP</li> </ul>
	FixedIP <sup>*1</sup>	--	次のプロパティのうち、どれか一つを必ず指定します。 <ul style="list-style-type: none"> <li>• netmask</li> <li>• prefixLength</li> <li>• FixedIP</li> <li>• ServiceIP</li> </ul>
	--	target	リクエスト先のノードの固有 IP アドレスを設定します。
		other	リクエスト先のノードの相手ノードの固有 IP アドレスを設定します。
		targetIPv6	リクエスト先のノードの固有 IP アドレスを IPv6 で設定します。
		otherIPv6	リクエスト先のノードの相手ノードの固有 IP アドレスを IPv6 で設定します。
	ServiceIP <sup>*1*2</sup>		--
	--	target	リクエスト先のノードの仮想 IP アドレスを設定します。空文字列を設定した場合は、設定されていた値が削除されます。
		other	リクエスト先のノードの相手ノードの仮想 IP アドレスを設定します。空文字列を設定した場合は、設定されていた値が削除されます。
		targetIPv6	リクエスト先のノードの仮想 IP アドレスを IPv6 で設定します。空文字列を設定した場合は、設定されていた値が削除されます。

プロパティ			説明	指定時の条件
		otherIPv6	リクエスト先のノードの相手ノードの仮想 IP アドレスを IPv6 で設定します。空文字列を設定した場合は、設定されていた値が削除されます。	固有 IP アドレスに IPv6 形式のアドレスを使用する場合は必須です。

注 \*1 次のネットワークの IP アドレスは指定しないでください。これらの IP アドレスはストレージシステム内部で予約されています。

- IPv4 の場合 :  
127.0.0.0 ~ 127.255.255.255
- IPv6 の場合 :  
::ffff: < IPv4 アドレス >, :: < IPv4 アドレス >, ::1/128, ::/0, ::/ 128, fe80::/10, ff00::/8
- 保守ポートに設定されている IP アドレスのネットワーク  
保守ポートに設定されている IP アドレスは pmctl コマンドで確認できます。
- ハートビートポートに設定されている IP アドレスのネットワーク  
ハートビートポートに設定されている IP アドレスは、ProcessingNode リソースを使用して確認してください。ProcessingNode リソースについては、「[3.10.2 ProcessingNode リソース](#)」を参照してください。

注 \*2 仮想 IP アドレスを指定しないで作成したインターフェースは、リソースグループの監視の対象外となります。仮想 IP アドレスを指定しないでインターフェースを作成するときの注意事項を次に示します。

- 仮想 IP アドレスが設定されていないインターフェースから、HVFP が提供する CIFS サービス、NFS サービス、FTP サービス、SFTP サービスおよび TFTP サービスを利用できません。
- 仮想 IP アドレスが設定されていないインターフェースでリンク障害が発生すると、SNMP トラップまたは E-mail 通知が出力されません。

NetworkInterfaces/ <ネットワークインターフェース名>リソースに POST メソッドをリクエストして、ネットワークインターフェースの情報を変更する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/NetworkInterfaces/<ネットワークインターフェース名>

### リクエスト XML の例 (IPv4 の場合)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NetworkInterface>
  <netmask>255.255.255.0</netmask>
  <FixedIP>
    <target>10.208.136.210</target>
    <other>10.208.136.211</other>
  </FixedIP>
  <ServiceIP>
    <target>10.208.136.212</target>
    <other>10.208.136.213</other>
  </ServiceIP>
</NetworkInterface>
```

### リクエスト XML の例 (IPv6 の場合)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NetworkInterface>
  <prefixLength>64</prefixLength>
  <FixedIP>
    <targetIPv6>2001:2c0:418:1::1</targetIPv6>
    <otherIPv6>2001:2c0:418:1::2</otherIPv6>
  </FixedIP>
```

```

    <ServiceIP>
      <targetIPv6>2001:2c0:418:1::3</targetIPv6>
      <otherIPv6>2001:2c0:418:1::4</otherIPv6>
    </ServiceIP>
  </NetworkInterface>

```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-66 : NetworkInterfaces/ <ネットワークインターフェース名>リソースに POST メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.8.3.3

## HEAD メソッドで指定したネットワークインターフェースが存在するかどうかを確認する

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-67 : NetworkInterfaces/ <ネットワークインターフェース名>リソースに HEAD メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	指定したネットワークインターフェースは使用できます。
その他	異常終了しました。「表 1-3 : 」を参照してください。

## 3.9

## NFSShare

### 3.9.1

## NFSShare リソースの概要

NFSShare リソースを使用して、NFS 共有の情報を管理できます。

NFSShare リソースに使用できる HTTP メソッドを次の表に示します。

**表 3-68 : NFSShare リソースに使用できる HTTP メソッド**

リソースの URI	HTTP メソッド	サポート構成	説明
/NFSShares	GET	クラスタ Virtual Server	ログインしているノード上または VirtualServer 上の NFS 共有の情報を取得します。
	PUT	クラスタ Virtual Server	NFS 共有を追加します。
/NFSShares/ < NFS 共有ディレク トリ名> *1	GET *2	クラスタ Virtual Server	指定した NFS 共有の情報を取得します。
	POST *2	クラスタ Virtual Server	指定した NFS 共有の情報を編集します。
	DELETE *2	クラスタ Virtual Server	指定した NFS 共有を削除します。
	HEAD *2	クラスタ Virtual Server	指定した NFS 共有が存在するかどうか確認します。

注 \*1 < NFS 共有ディレクトリ名>の個所には、共有ディレクトリの最下層にあるディレクトリの名称を指定します。

注 \*2 クエリパラメーター SHARE-DIRECTORY-NAME を使用して、共有ディレクトリのパス名を必ず指定してください。

## 3.9.2 NFSShare リソース

この項では、NFSShare リソースの使い方について説明します。

### 3.9.2.1 GET メソッドですべての NFS 共有の情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

表 3-69 : NFSShare リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ				出力される 個数	サポート構成	verbose が false の場合の 表示		
NFSShares				1	クラスタ Virtual Server	○		
--	NFSShare			0, 1 ～ n	クラスタ Virtual Server	○		
	--	exportPoint		1	クラスタ Virtual Server	○		
		anonymousUID		1	クラスタ Virtual Server	○		
		anonymousGID		1	クラスタ Virtual Server	○		
		isPortRestrictionEnabled		1	クラスタ Virtual Server	○		
		isSubtreeCheckEnabled		1	クラスタ Virtual Server	○		
		isAccessCheckWithLockRequestEnabled		1	クラスタ Virtual Server	○		
		aclType		1	クラスタ Virtual Server	×		
		maxRwSize		1	クラスタ Virtual Server	○		
		AllowedHosts		1	クラスタ Virtual Server	○		
		--	AllowedHost		1 ～ n	クラスタ Virtual Server	○	
			--	hostOrAddress		1	クラスタ Virtual Server	○
				isReadOnly		1	クラスタ Virtual Server	○
				synchronousWritingSetting		1	クラスタ Virtual Server	○
				anonymousMapping		1	クラスタ Virtual Server	○
				securityFlavor		1	クラスタ Virtual Server	○

(凡例) n : 2 以上の整数を示します。○ : 表示されます。× : 表示されません。

NFSShare リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

表 3-70 : NFSShare リソースに GET メソッドをリクエストしたときのプロパティ

プロパティ			説明
NFSShares			--
--	NFSShare		--
	--	exportPoint	共有ディレクトリが絶対パスで表示されます。
	--	anonymousUID	匿名ユーザーとしてアクセスするときに使用するユーザー ID が表示されます。
	--	anonymousGID	匿名ユーザーとしてアクセスするときに使用するグループ ID が表示されます。
	--	isPortRestrictionEnabled	1024 番より小さなインターネットポートから発信されたリクエストだけを受け付けるかどうかが表示されます。 true 1024 番より小さなインターネットポートから発信されたリクエストだけを受け付ける設定の場合に表示されます。 false すべてのインターネットポートから発信されたリクエストを受け付ける設定の場合に表示されます。
	--	isSubtreeCheckEnabled	アクセス対象のファイルやディレクトリに対する権限をチェックするだけでなく、共有ディレクトリからアクセス対象のファイルやディレクトリが存在するディレクトリまでの全ディレクトリ（サブツリー）に実行権限（x）があることもチェックするかどうかが表示されます。 true チェックをする設定の場合に表示されます。 false チェックをしない設定の場合に表示されます。
	--	isAccessCheckWithLockRequestEnabled	ロック要求があった場合に、権限チェックするかどうかの設定が表示されます。 true 権限チェックをする設定の場合に表示されます。 false 権限チェックをしない設定の場合に表示されます。
	--	aclType	ファイルシステムの ACL タイプが表示されます。 Advanced ACL Advanced ACL のときに表示されます。 Classic ACL Classic ACL のときに表示されます。 Unknown ACL タイプを取得できなかったときに表示されます。
	--	maxRwSize	NFS 共有に対する最大転送長が表示されます。 最大転送長が設定されていない場合は、空文字列が表示されます。
	--	AllowedHosts	--

プロパティ				説明
		--	AllowedHost	--
			-- hostOrAddress	NFS 共有を公開しているホスト名またはネットワーク名が表示されます。
			isReadOnly	NFS 共有に対するアクセス権が表示されます。 true 読み取りだけが許可されている場合に 표시됩니다。 false 読み取りおよび書き込みが許可されている場合に 표시됩니다。
			synchronousWritingSetting	isReadOnly プロパティが false の場合、同期書き込みを最適化するかどうかについての情報が表示されます。表示される値は、isReadOnly プロパティに false を設定している場合だけ有効です。 rw_sync 同期書き込みを最適化しない設定の場合に表示されます。 rw_sync_opt 同期書き込みを最適化する設定の場合に表示されます。
			anonymousMapping	匿名ユーザーとしてマッピングするユーザーが表示されます。 everyone すべてのユーザーを匿名ユーザーにマッピングする設定の場合に表示されます。 root_only root ユーザーだけを匿名ユーザーにマッピングする設定の場合に表示されます。 none 匿名ユーザーのマッピングを行わない設定の場合に表示されます。
			securityFlavor	NFS 共有の公開先に設定されているセキュリティフレーバーが表示されます。複数のセキュリティフレーバーが設定されている場合は、ピリオド (.) で区切って表示されます。 sys UNIX (AUTH_SYS) 認証を適用する場合に表示されます。 krb5 Kerberos 認証を適用する場合に表示されます。 krb5i Kerberos 認証に加えてデータ完全性機能を適用する場合に表示されます。 krb5p Kerberos 認証に加えてデータ完全性機能およびプライバシー機能を適用する場合に表示されます。

NFSShare リソースに GET メソッドをリクエストして、NFS 共有の情報を取得する例を次に示します。

#### リクエスト先の URI

<https://<ホスト名またはIPアドレス>:9090/mapi/NFSShares>



### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NFSShares>
  <NFSShare>
    <exportPoint>/mnt/fs/nfs</exportPoint>
    <anonymousUID>22222</anonymousUID>
    <anonymousGID>22222</anonymousGID>
    <isPortRestrictionEnabled>true</isPortRestrictionEnabled>
    <isSubtreeCheckEnabled>true</isSubtreeCheckEnabled>
    <isAccessCheckWithLockRequestEnabled>true</
isAccessCheckWithLockRequestEnabled>
    <maxRwSize>5</maxRwSize>
    <AllowedHosts>
      <AllowedHost>
        <hostOrAddress>*</hostOrAddress>
        <isReadOnly>true</isReadOnly>
        <synchronousWritingSetting>rw_sync</
synchronousWritingSetting>
        <anonymousMapping>root_only</anonymousMapping>
        <securityFlavor>sys</securityFlavor>
      </AllowedHost>
    </AllowedHosts>
  </NFSShare>
</NFSShares>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-71 : NFSShare リソースに GET メソッドをリクエストしたときのステータスコード

ステータスコード	状態
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.9.2.2

### PUT メソッドで NFS 共有を追加する

NFSShare リソースに PUT メソッドをリクエストするときの XML の構造を次の表に示します。

表 3-72 : NFSShare リソースに PUT メソッドをリクエストするときの XML の構造

プロパティ		データ型	指定できる個数	サポート構成
NFSShare		--	1	クラスタ Virtual Server
--	exportPoint	String	1	クラスタ Virtual Server
	anonymousUID	Integer	0, 1	クラスタ Virtual Server
	anonymousGID	Integer	0, 1	クラスタ Virtual Server
	isPortRestrictionEnabled	Boolean	0, 1	クラスタ Virtual Server
	isSubtreeCheckEnabled	Boolean	0, 1	クラスタ Virtual Server
	isAccessCheckWithLockRequest Enabled	Boolean	0, 1	クラスタ Virtual Server
	maxRwSize	Integer	0, 1	クラスタ Virtual Server

プロパティ				データ型	指定できる個数	サポート構成	
	AllowedHosts			--	1	クラスタ Virtual Server	
	--	AllowedHost		--	0, 1 ～ n	クラスタ Virtual Server	
		--	hostOrAddress		String	1	クラスタ Virtual Server
			isReadOnly		Boolean	0, 1	クラスタ Virtual Server
			synchronousWritingSetting		String	0, 1	クラスタ Virtual Server
			anonymousMapping		String	0, 1	クラスタ Virtual Server
			securityFlavor		String	0, 1	クラスタ Virtual Server
			DirectorySetup			--	0, 1
	--	userName		String	0, 1	クラスタ Virtual Server	
		groupName		String	0, 1	クラスタ Virtual Server	
		isStickyBit		Boolean	0, 1	クラスタ Virtual Server	
		ownerPermission		String	0, 1	クラスタ Virtual Server	
		groupPermission		String	0, 1	クラスタ Virtual Server	
		otherPermission		String	0, 1	クラスタ Virtual Server	
	DirectoryACLSettings			--	0, 1	クラスタ Virtual Server	
	--	DirectoryACLSetting		--	0, 1 ～ n	クラスタ Virtual Server	
		--	userGroupAccess		String	1	クラスタ Virtual Server
			name		String	1	クラスタ Virtual Server
			accountType		String	1	クラスタ Virtual Server
			applicationDestination		String	1	クラスタ Virtual Server
			inheritanceRange		String	1	クラスタ Virtual Server
			accessPermission		String	1	クラスタ Virtual Server

(凡例) n : 2 以上の整数を示します。

NFSShare リソースに PUT メソッドをリクエストするときに指定するプロパティについて次の表に示します。

表 3-73 : NFSShare リソースに PUT メソッドをリクエストするときのプロパティ

プロパティ		説明	指定時の条件
NFSShare		--	--
--	exportPoint	<p>NFS 共有を作成するディレクトリを 6 文字以上 63 文字以内の絶対パスで指定します。差分スナップショットの自動作成スケジュール（共有の自動作成）を運用する場合は、絶対パスが 48 文字以内になるようにしてください。</p> <p>このプロパティに「/mnt/&lt;ファイルシステム名&gt;」で始まるパスを指定することで、ファイルシステムと NFS 共有が対応します。</p> <p>例えば、filesystem01 ファイルシステムの share01 ディレクトリにファイル共有を作成する場合は、/mnt/filesystem01/share01 と指定します。</p> <p>指定できる文字は、英数字、ハイフン (-)、ピリオド (.), 斜線 (/) およびアンダーライン (_) です。マルチバイト文字は指定できません。</p> <p>共有ディレクトリはクラスタ内で一意である必要があります。Virtual Server 上でファイル共有を追加する場合は、Virtual Server 内で重複しない名称を指定します。なお、「.conflict」, 「.conflict_longpath」, 「.snaps」, 「.history」および「.lost+found」というディレクトリ名は指定できません。また、「.arc」, 「.system_gi」, 「.system_reorganize」, 「.backupdates」, 「.temp_backupdates」および「lost+found」は、ファイルシステム直下のディレクトリの名称として指定できません。シンボリックリンクを含むパスは指定できません。末尾に指定した斜線およびスペースは削除されます。</p>	必須
	anonymousUID	<p>匿名ユーザーとしてアクセスするときに使用するユーザー ID を設定します。</p> <p>0 ～ 65535 の範囲で指定します。省略した場合は、65534 が設定されます。</p>	任意
	anonymousGID	<p>匿名ユーザーとしてアクセスするときに使用するグループ ID を設定します。</p> <p>0 ～ 65535 の範囲で指定します。省略した場合は、65534 が設定されます。</p>	任意
	isPortRestrictionEnabled	<p>1024 番より小さなインターネットポートから発信されたリクエストだけを受け付けるかどうかを指定します。</p> <p>true</p> <p>1024 番より小さなインターネットポートから発信されたリクエストだけを受け付ける場合に指定します。</p> <p>false*</p> <p>すべてのインターネットポートから発信されたリクエストを受け付ける場合に指定します。</p>	任意

プロパティ			説明	指定時の条件
	isSubtreeCheckEnabled		共有ディレクトリからアクセス対象のファイルやディレクトリが存在するディレクトリまでのすべてのディレクトリ（サブツリー）に、実行権限（x）があることをチェックするかどうかを設定します。 true サブツリーをチェックする場合に指定します。 false* サブツリーをチェックしない場合に指定します。	任意
	maxRwSize		NFS 共有に対する最大転送長を KB 単位で 4 ～ 1024 の範囲で指定します。 省略した場合、値は設定されません。	任意
	AllowedHosts		--	任意
	--	AllowedHost	--	任意
	--	hostOrAddress	NFS 共有の公開先を設定します。 255 文字以内で指定します。ハイフン（-）は文字列の先頭には指定できません。ホストの別名を指定する場合は、正式名も 255 文字以内である必要があります。 なお、指定したホスト名またはネットワークアドレスの長さ（指定長 +5 バイト）の合計が、1,258 バイト未満である必要があります。 ホスト名、IP アドレスのほか、次の形式で指定できます。 <ul style="list-style-type: none"> <li>ネットグループ NIS のネットグループを指定します。 例えば、「@group」のように指定した場合、ネットグループのメンバーのうち、ホストの部分だけが取り出されます。</li> <li>IP ネットワーク サブネットワーク内のすべてのホストに公開する場合、IP アドレスとネットマスクを次の形式で指定します。 &lt;アドレス&gt; / &lt;ネットマスク&gt; ネットマスクは 10 進数をピリオド（.）で区切って指定することも、連続するマスクの長さで指定することもできます。</li> <li>DNS ドメイン NFS クライアントが属する DNS ドメインの名称を、先頭にピリオド（.）を付けて指定します。 例：.example.com</li> <li>ワイルドカード すべてのホストを指定するときに、アスタリスク（*）をワイルドカードとして指定できます。</li> </ul>	必須

プロパティ			説明	指定時の条件
			<p>なお、NFS クライアントマシンに、HVFP と通信するネットワークインターフェースが複数ある場合は、次のどれかの方法で公開先を指定してください。</p> <ul style="list-style-type: none"> <li>・ ワイルドカード (*) を使用する</li> <li>・ NFS クライアント側で使用するすべてのネットワークインターフェースの IP アドレスを指定する</li> <li>・ NFS クライアント側で使用するすべてのネットワークインターフェースに対応するホスト名を指定する</li> <li>・ NFS クライアント側で使用するすべてのネットワークインターフェースの IP アドレスを含む IP ネットワークを指定する</li> <li>・ NFS クライアント側で使用するすべてのネットワークインターフェースに対応するホスト名を含むネットワークグループを指定する</li> <li>・ NFS クライアント側で使用するすべてのネットワークインターフェースに対応するホスト名を含む DNS ドメインを指定する</li> </ul>	
		isReadOnly	<p>NFS 共有の読み取りだけを許可するかどうかを設定します。</p> <p>true 読み取り専用にする場合に指定します。</p> <p>false* 読み取り専用にしない場合に指定します。</p>	任意
		synchronousWriting Setting	<p>同期書き込みを最適化するかどうかを設定します。isReadOnly プロパティに false を設定した場合だけ、有効です。</p> <p>rw_sync* 同期書き込みを最適化しない場合に指定します。</p> <p>rw_sync_opt 同期書き込みを最適化する場合に指定します。</p>	任意
		anonymousMapping	<p>匿名ユーザーへのマッピングを使用する場合に設定します。</p> <p>everyone すべてのユーザーを匿名ユーザーにマッピングする場合に指定します。</p> <p>root_only* root ユーザーだけを匿名ユーザーにマッピングする場合に指定します。</p> <p>none 匿名ユーザーへのマッピングを行わない場合に指定します。</p>	任意

プロパティ			説明	指定時の条件
		securityFlavor	<p>セキュリティフレーバーを設定します。</p> <p>複数指定する場合はピリオド (.) で区切ってください。ただし、default は複数指定できません。</p> <p>sys UNIX (AUTH_SYS) 認証を適用する場合に指定します。</p> <p>krb5 Kerberos 認証を適用する場合に指定します。</p> <p>krb5i Kerberos 認証に加えてデータ完全性機能を適用する場合に指定します。</p> <p>krb5p Kerberos 認証に加えてデータ完全性機能およびプライバシー機能を適用する場合に指定します。</p> <p>default* GUI の [NFS Service Management] ページで設定されているセキュリティフレーバーの値を適用する場合に指定します。この値は、ほかの値と同時に指定できません。 なお、メソッドを実行した時点の設定値が適用されます。</p>	任意
		DirectorySetup	<p>共有の作成と同時に、共有ディレクトリを作成する場合に指定します。</p> <p>このプロパティで設定した内容を確認するには、作成した共有ディレクトリにクライアントからアクセスできるかどうかを確認してください。</p>	任意
	--	userName	<p>作成するディレクトリのオーナーのユーザー名またはユーザー ID を設定します。</p> <p>ユーザー ID で設定する場合は、0 ～ 2147483147 の範囲で指定してください。ただし、「0」以外の ID については、先頭に 0 を指定できません。</p> <p>Windows ドメインのビルトインユーザーは指定できません。</p> <p>省略した場合、userName プロパティおよび groupName プロパティには「root」が設定され、isStickyBit プロパティ、ownerPermission プロパティ、groupPermission プロパティ、および otherPermission プロパティのアクセス権は設定されません。</p>	DirectorySetup プロパティを設定する場合に指定できます。
		groupName	<p>ディレクトリを所有するグループの名称または ID を指定します。</p> <p>グループ ID で設定する場合は、0 ～ 2147483147 の範囲で指定してください。ただし、「0」以外の ID については、先頭に 0 を指定できません。</p> <p>Windows ドメインのビルトイングループは指定できません。</p> <p>省略した場合、userName プロパティおよび groupName プロパティには「root」が設定され、isStickyBit プロパティ、ownerPermission プロパティ、groupPermission プロパティ、および otherPermission プロパティのアクセス権は設定されません。</p>	DirectorySetup プロパティを設定する場合に指定できます。

プロパティ		説明	指定時の条件
	isStickyBit	<p>スティッキービットを使用するかどうかを設定します。</p> <p>このプロパティの指定は、userName プロパティおよび groupName プロパティを設定する場合にだけ有効です。</p> <p>true</p> <p>スティッキービットを使用する場合に指定します。</p> <p>false*</p> <p>スティッキービットを使用しない場合に指定します。</p>	DirectorySetup プロパティを設定する場合に指定できます。
	ownerPermission	<p>ディレクトリへのオーナーのアクセス権を設定します。</p> <p>このプロパティの指定は、userName プロパティおよび groupName プロパティを設定する場合にだけ有効です。</p> <p>省略した場合、ownerPermission プロパティにアクセス権は設定されませんが、isStickyBit プロパティ、groupPermission プロパティ、および otherPermission プロパティをすべて省略すると、ownerPermission プロパティには rw が設定されます。</p> <p>rw</p> <p>読み取りおよび書き込みを許可する場合に指定します。</p> <p>ro</p> <p>読み取りだけを許可する場合に指定します。</p> <p>none</p> <p>読み取りおよび書き込みを許可しない場合に指定します。</p>	DirectorySetup プロパティを設定する場合に指定できます。
	groupPermission	<p>ディレクトリへのグループのアクセス権を設定します。</p> <p>このプロパティの指定は、userName プロパティおよび groupName プロパティを設定する場合にだけ有効です。</p> <p>省略した場合、groupPermission プロパティにアクセス権は設定されませんが、isStickyBit プロパティ、ownerPermission プロパティ、および otherPermission プロパティをすべて省略すると、groupPermission プロパティには ro が設定されます。</p> <p>設定するアクセス権については、ownerPermission プロパティを参照してください。</p>	DirectorySetup プロパティを設定する場合に指定できます。

プロパティ			説明	指定時の条件
		otherPermission	ディレクトリへのその他のアクセス権を設定します。 このプロパティの指定は、userName プロパティおよび groupName プロパティを設定する場合にだけ有効です。 省略した場合、otherPermission プロパティにアクセス権は設定されませんが、isStickyBit プロパティ、ownerPermission プロパティ、および groupPermission プロパティをすべて省略すると、otherPermission プロパティには ro が設定されます。 設定するアクセス権については、ownerPermission プロパティを参照してください。	DirectorySetup プロパティを設定する場合に指定できます。
		DirectoryACLSettings	共有ディレクトリに ACL を設定する場合に指定します。 このプロパティで設定した内容を確認するには、作成した共有ディレクトリにクライアントからアクセスできるかどうかを確認してください。	isClassicAcl プロパティが false の場合に指定できます。
	--	DirectoryACLSetting	--	DirectoryACLSettings プロパティを設定する場合に必須です。
	--	userGroupAccess	ACL のユーザーまたはグループのアクセス権を設定します。 allow 指定したユーザーまたはグループに許可のアクセス権を与えます。ユーザーまたはグループの ACE にアクセス権が追加され、すでに ACE が設定されている場合は、アクセス権が統合されます。同一ユーザーに対して複数の ACE が設定されている場合は、一つの ACE に統合されます。 replace 指定したユーザーまたはグループに許可のアクセス権を与える場合に指定します。同一ユーザーまたはグループの ACE は破棄され、指定した ACE に置き換えられます。 deny 指定したユーザーまたはグループに拒否のアクセス権を与える場合に指定します。同一ユーザーまたはグループの ACE は破棄され、指定した ACE に置き換えられます。	DirectoryACLSetting プロパティを設定する場合に必須です。
		name	ACL のユーザー名またはグループ名を設定します。	DirectoryACLSetting プロパティを設定する場合に必須です。



プロパティ			説明	指定時の条件
		accountType	ACL のアカウント種別を設定します。 u ユーザーにアクセス権を与える場合に指定します。 g グループにアクセス権を与える場合に指定します。	DirectoryACLSetting プロパティを設定する場合に必須です。
		applicationDestination	ACE が適用される範囲を設定します。 o ディレクトリだけに適用する場合に指定します。 a ディレクトリ、サブディレクトリおよびファイルに適用する場合に指定します。 f ディレクトリとサブディレクトリに適用する場合に指定します。 e ディレクトリとファイルに適用する場合に指定します。 s サブディレクトリとファイルに適用する場合に指定します。 u サブディレクトリだけに適用する場合に指定します。 i ファイルだけに適用する場合に指定します。	DirectoryACLSetting プロパティを設定する場合に必須です。
		inheritanceRange	ACE の継承範囲を設定します。 o 直下にあるディレクトリやファイルにだけ継承する場合に指定します。 a 下位にあるすべてのディレクトリおよびファイルに継承する場合に指定します。	DirectoryACLSetting プロパティを設定する場合に必須です。

プロパティ			説明	指定時の条件
		accessPermission	<p>アクセス権を設定します。</p> <p>複数のアクセス権を設定する場合は、コロン二つ (:) で区切って指定してください。</p> <p>f 「フルコントロール」を許可する場合に指定します。</p> <p>s 「フォルダのスキャン/ファイルの実行」を許可する場合に指定します。</p> <p>r 「フォルダの一覧/データの読み取り」を許可する場合に指定します。</p> <p>a 「属性の読み取り」を許可する場合に指定します。</p> <p>e 「拡張属性の読み取り」を許可する場合に指定します。</p> <p>w 「ファイルの作成/データの書き込み」を許可する場合に指定します。</p> <p>p 「フォルダの作成/データの追加」を許可する場合に指定します。</p> <p>t 「属性の書き込み」を許可する場合に指定します。</p> <p>x 「拡張属性の書き込み」を許可する場合に指定します。</p> <p>l 「サブフォルダとファイルの削除」を許可する場合に指定します。</p> <p>d 「削除」を許可する場合に指定します。</p> <p>c 「アクセス許可の読み取り」を許可する場合に指定します。</p> <p>h 「アクセス許可の変更」を許可する場合に指定します。</p> <p>g 「所有権の取得」を許可する場合に指定します。</p>	DirectoryACLSetting プロパティを設定する場合に必須です。

注 \* プロパティを省略した場合のデフォルト値です。

NFSShare リソースに PUT メソッドをリクエストして、NFS 共有を作成する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/NFSShares

#### リクエスト XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NFSShare>
  <exportPoint>/mnt/fs/nfs</exportPoint>
  <anonymousUID>22222</anonymousUID>
  <anonymousGID>22222</anonymousGID>
</NFSShare>
```

```

    <isPortRestrictionEnabled>true</isPortRestrictionEnabled>
    <isSubtreeCheckEnabled>true</isSubtreeCheckEnabled>
    <isAccessCheckWithLockRequestEnabled>true</
isAccessCheckWithLockRequestEnabled>
    <maxRwSize>5</maxRwSize>
    <AllowedHosts>
      <AllowedHost>
        <hostOrAddress>*</hostOrAddress>
        <isReadOnly>true</isReadOnly>
        <synchronousWritingSetting>rw_sync</synchronousWritingSetting>
        <anonymousMapping>root_only</anonymousMapping>
        <securityFlavor>sys</securityFlavor>
      </AllowedHost>
    </AllowedHosts>
    <DirectorySetup>
      <userName>22222</userName>
      <groupName>11111</groupName>
      <isStickyBit>true</isStickyBit>
      <ownerPermission>rw</ownerPermission>
      <groupPermission>rw</groupPermission>
      <otherPermission>rw</otherPermission>
    </DirectorySetup>
    <DirectoryACLSettings>
      <DirectoryACLSetting>
        <userGroupAccess>allow</userGroupAccess>
        <name>Everyone</name>
        <accountType>g</accountType>
        <applicationDestination>a</applicationDestination>
        <inheritanceRange>a</inheritanceRange>
        <accessPermission>r</accessPermission>
      </DirectoryACLSetting>
    </DirectoryACLSettings>
  </NFSShare>

```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-74 : NFSShare リソースに PUT メソッドをリクエストしたときのステータスコード**

ステータスコード	状態
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.9.3 NFSShares/ < NFS 共有ディレクトリ名 > リソース

この項では、NFSShares/ < NFS 共有ディレクトリ名 > リソースの使い方について説明します。

### 3.9.3.1

## GET メソッドで指定した NFS 共有の情報を取得する

クエリパラメーター SHARE-DIRECTORY-NAME を使用して、共有ディレクトリのパス名を必ず指定してください。

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

**表 3-75：NFSShares/ < NFS 共有ディレクトリ名 > リソースに GET メソッドをリクエストしたときのレスポンス XML の構造**

プロパティ			出力される個数	サポート構成	verbose が false の場合の表示
NFSShare			1	クラスタ Virtual Server	○
--	exportPoint		1	クラスタ Virtual Server	○
	anonymousUID		1	クラスタ Virtual Server	○
	anonymousGID		1	クラスタ Virtual Server	○
	isPortRestrictionEnabled		1	クラスタ Virtual Server	○
	isSubtreeCheckEnabled		1	クラスタ Virtual Server	○
	isAccessCheckWithLockRequest Enabled		1	クラスタ Virtual Server	○
	aclType		1	クラスタ Virtual Server	×
	maxRwSize		1	クラスタ Virtual Server	○
	AllowedHosts		1	クラスタ Virtual Server	○
	--	AllowedHost	1 ~ n	クラスタ Virtual Server	○
		hostOrAddress	1	クラスタ Virtual Server	○
		isReadOnly	1	クラスタ Virtual Server	○
		synchronousWriti ngSetting	1	クラスタ Virtual Server	○
		anonymousMapping	1	クラスタ Virtual Server	○
		securityFlavor	1	クラスタ Virtual Server	○

(凡例) n : 2 以上の整数を示します。○ : 表示されます。× : 表示されません。

NFSShares/ < NFS 共有ディレクトリ名 > リソースに GET メソッドをリクエストしたときに表示されるプロパティについては、「[表 3-70：NFSShare リソースに GET メソッドをリクエストしたときのプロパティ](#)」を参照してください。

NFSShares/ < NFS 共有ディレクトリ名 > リソースに GET メソッドをリクエストして、指定した NFS 共有の情報を取得する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/NFSShares/<NFS共有ディレクトリ名>

### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NFSShare>
  <exportPoint>/mnt/fs/nfs</exportPoint>
  <anonymousUID>22222</anonymousUID>
  <anonymousGID>22222</anonymousGID>
  <isPortRestrictionEnabled>true</isPortRestrictionEnabled>
  <isSubtreeCheckEnabled>true</isSubtreeCheckEnabled>
  <isAccessCheckWithLockRequestEnabled>true</
isAccessCheckWithLockRequestEnabled>
  <maxRwSize>5</maxRwSize>
  <AllowedHosts>
    <AllowedHost>
      <hostOrAddress>*</hostOrAddress>
      <isReadOnly>true</isReadOnly>
      <synchronousWritingSetting>rw_sync</synchronousWritingSetting>
      <anonymousMapping>root_only</anonymousMapping>
      <securityFlavor>sys</securityFlavor>
    </AllowedHost>
  </AllowedHosts>
</NFSShare>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-76 : NFSShares/ < NFS 共有ディレクトリ名>リソースに GET メソッドをリクエストしたときのステータスコード**

ステータスコード	状態
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.9.3.2

### POST メソッドで NFS 共有の情報を編集する

クエリパラメーター SHARE-DIRECTORY-NAME を使用して、共有ディレクトリのパス名を必ず指定してください。

NFSShares/ < NFS 共有ディレクトリ名>リソースに POST メソッドをリクエストするときの XML の構造を次の表に示します。

表 3-77 : NFSShares/ < NFS 共有ディレクトリ名 > リソースに POST メソッドをリクエストするときの XML の構造

プロパティ			データ型	指定できる個数	サポート構成
NFSShare			--	1	クラスタ Virtual Server
--	anonymousUID		Integer	0, 1	クラスタ Virtual Server
	anonymousGID		Integer	0, 1	クラスタ Virtual Server
	isPortRestrictionEnabled		Boolean	0, 1	クラスタ Virtual Server
	isSubtreeCheckEnabled		Boolean	0, 1	クラスタ Virtual Server
	isAccessCheckWithLockRequest Enabled		Boolean	0, 1	クラスタ Virtual Server
	maxRwSize		Integer	0, 1	クラスタ Virtual Server
	AllowedHosts		--	1	クラスタ Virtual Server
--	--	AllowedHost	--	1 ～ n	クラスタ Virtual Server
		hostOrAddress	String	1	クラスタ Virtual Server
		isReadOnly	Boolean	0, 1	クラスタ Virtual Server
		synchronousWriti ngSetting	String	0, 1	クラスタ Virtual Server
		anonymousMapping	String	0, 1	クラスタ Virtual Server
		securityFlavor	String	0, 1	クラスタ Virtual Server

(凡例) n : 2 以上の整数を示します。

NFSShares/ < NFS 共有ディレクトリ名 > リソースに POST メソッドをリクエストするときに指定するプロパティについて次の表に示します。

表 3-78 : NFSShares/ < NFS 共有ディレクトリ名 > リソースに POST メソッドをリクエストするときのプロパティ

プロパティ		説明	指定時の条件
NFSShare		--	--
--	anonymousUID	匿名ユーザーとしてアクセスするときに使用するユーザー ID を設定します。 0 ~ 65535 の範囲で指定します。	任意
	anonymousGID	匿名ユーザーとしてアクセスするときに使用するグループ ID を設定します。 0 ~ 65535 の範囲で指定します。	任意

プロパティ			説明	指定時の条件
	isPortRestrictionEnabled		<p>1024 番より小さなインターネットポートから発信されたリクエストだけを受け付けるかどうかを指定します。</p> <p>true</p> <p>1024 番より小さなインターネットポートから発信されたリクエストだけを受け付ける場合に指定します。</p> <p>false</p> <p>すべてのインターネットポートから発信されたリクエストを受け付ける場合に指定します。</p>	任意
	isSubtreeCheckEnabled		<p>共有ディレクトリからアクセス対象のファイルやディレクトリが存在するディレクトリまでのすべてのディレクトリ（サブツリー）に、実行権限（x）があることをチェックするかどうかを設定します。</p> <p>true</p> <p>サブツリーをチェックする場合に指定します。</p> <p>false</p> <p>サブツリーをチェックしない場合に指定します。</p>	任意
	isAccessCheckWithLockRequestEnabled		<p>ロック要求があった場合に、権限チェックするかどうかを設定します。</p> <p>true</p> <p>権限チェックする場合に指定します。</p> <p>false</p> <p>権限チェックしない場合に指定します。</p>	任意
	maxRwSize		NFS 共有に対する最大転送長を KB 単位で 4 ～ 1024 の範囲で指定します。	任意
	AllowedHosts		--	--
	--	AllowedHost	--	--
	--	hostOrAddress	<p>NFS 共有の公開先を設定します。</p> <p>255 文字以内で指定します。ハイフン（-）は文字列の先頭には指定できません。ホストの別名を指定する場合は、正式名も 255 文字以内である必要があります。</p> <p>なお、指定したホスト名またはネットワークアドレスの長さ（指定長 +5 バイト）の合計が、1,258 バイト未満である必要があります。</p> <p>ホスト名、IP アドレスのほか、次の形式で指定できます。</p> <ul style="list-style-type: none"> <li>ネットグループ</li> </ul> <p>NIS のネットグループを指定します。</p> <p>例えば、「@group」のように指定した場合、ネットグループのメンバーのうち、ホストの部分だけが取り出されます。</p>	必須

プロパティ				説明	指定時の条件
				<ul style="list-style-type: none"> <li>IP ネットワーク サブネットワーク内のすべてのホストに公開する場合、IP アドレスとネットマスクを次の形式で指定します。 &lt;アドレス&gt; / &lt;ネットマスク&gt; ネットマスクは 10 進数をピリオド (.) で区切って指定することも、連続するマスクの長さで指定することもできます。</li> <li>DNS ドメイン NFS クライアントが属する DNS ドメインの名称を、先頭にピリオド (.) を付けて指定します。 例: .example.com</li> <li>ワイルドカード すべてのホストを指定するときに、アスタリスク (*) をワイルドカードとして指定できます。</li> </ul> <p>なお、NFS クライアントマシンに、HVFP と通信するネットワークインターフェースが複数ある場合は、次のどれかの方法で公開先を指定してください。</p> <ul style="list-style-type: none"> <li>ワイルドカード (*) を使用する</li> <li>NFS クライアント側で使用するすべてのネットワークインターフェースの IP アドレスを指定する</li> <li>NFS クライアント側で使用するすべてのネットワークインターフェースに対応するホスト名を指定する</li> <li>NFS クライアント側で使用するすべてのネットワークインターフェースの IP アドレスを含む IP ネットワークを指定する</li> <li>NFS クライアント側で使用するすべてのネットワークインターフェースに対応するホスト名を含むネットグループを指定する</li> <li>NFS クライアント側で使用するすべてのネットワークインターフェースに対応するホスト名を含む DNS ドメインを指定する</li> </ul>	
			isReadOnly	<p>NFS 共有の読み取りだけを許可するかどうかを設定します。</p> <p>true 読み取り専用にする場合に指定します。</p> <p>false 読み取り専用にしない場合に指定します。</p>	任意



プロパティ				説明	指定時の条件
			synchronousWritingSetting	同期書き込みを最適化するかどうかを設定します。 isReadOnly プロパティに false を設定した場合だけ、有効です。 rw_sync 同期書き込みを最適化しない場合に指定します。 rw_sync_opt 同期書き込みを最適化する場合に指定します。	任意
			anonymousMapping	匿名ユーザーへのマッピングを使用する場合に設定します。 everyone すべてのユーザーを匿名ユーザーにマッピングする場合に指定します。 root_only root ユーザーだけを匿名ユーザーにマッピングする場合に指定します。 none 匿名ユーザーへのマッピングを行わない場合に指定します。	任意
			securityFlavor	セキュリティフレーバーを設定します。 sys UNIX (AUTH_SYS) 認証を適用する場合に指定します。 krb5 Kerberos 認証を適用する場合に指定します。 krb5i Kerberos 認証に加えてデータ完全性機能を適用する場合に指定します。 krb5p Kerberos 認証に加えてデータ完全性機能およびプライバシー機能を適用する場合に指定します。 default GUI の [NFS Service Management] ページで設定されているセキュリティフレーバーの値を適用する場合に指定します。	任意

NFSShares/ < NFS 共有ディレクトリ名 > リソースに POST メソッドをリクエストして、NFS 共有の情報を編集する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/NFSShares/<NFS共有ディレクトリ名>

### リクエスト XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NFSShare>
  <anonymousUID>33333</anonymousUID>
  <anonymousGID>33333</anonymousGID>
  <isPortRestrictionEnabled>>false</isPortRestrictionEnabled>
  <isSubtreeCheckEnabled>>false</isSubtreeCheckEnabled>
  <isAccessCheckWithLockRequestEnabled>>false</isAccessCheckWithLockRequestEnabled>
  <maxRwSize>6</maxRwSize>
</NFSShare>
```

```

    <AllowedHosts>
      <AllowedHost>
        <hostOrAddress>*</hostOrAddress>
        <isReadOnly>>false</isReadOnly>
        <synchronousWritingSetting>rw_sync</synchronousWritingSetting>
        <anonymousMapping>root_only</anonymousMapping>
        <securityFlavor>sys</securityFlavor>
      </AllowedHost>
    </AllowedHosts>
  </NFSShare>

```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-79 : NFSShares/ < NFS 共有ディレクトリ名 > リソースに POST メソッドをリクエストしたときのステータスコード**

ステータスコード	状態
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.9.3.3 DELETE メソッドで指定した NFS 共有を削除する

クエリパラメーター SHARE-DIRECTORY-NAME を使用して、共有ディレクトリのパス名を必ず指定してください。

NFS 共有を削除しても、共有ディレクトリは削除されません。

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-80 : NFSShares/ < NFS 共有ディレクトリ名 > リソースに DELETE メソッドをリクエストしたときのステータスコード**

ステータスコード	状態
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.9.3.4 HEAD メソッドで指定した NFS 共有が存在するかどうか確認する

クエリパラメーター SHARE-DIRECTORY-NAME を使用して、共有ディレクトリのパス名を必ず指定してください。

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-81 : NFSShares/ < NFS 共有ディレクトリ名 > リソースに HEAD メソッドをリクエストしたときのステータスコード**

ステータスコード	状態
200	指定した NFS 共有は利用できます。
その他	異常終了しました。「表 1-3 : 」を参照してください。

## 3.10 ProcessingNode

### 3.10.1 ProcessingNode リソースの概要

ProcessingNode リソースを使用して、クラスタやノードの情報を取得したり、ホスト名を設定したりできます。

ProcessingNode リソースに使用できる HTTP メソッドを次の表に示します。

表 3-82 : ProcessingNode リソースに使用できる HTTP メソッド

リソースの URI	HTTP メソッド	サポート構成	説明
/ProcessingNode	GET	クラスタ	クラスタおよびノードの情報を取得します。

## 3.10.2 ProcessingNode リソース

この項では、ProcessingNode リソースの使い方について説明します。

### 3.10.2.1 GET メソッドでクラスタおよびノードの情報を取得する

ここでは、GET メソッドを使ってクラスタおよびノードの情報を取得する方法について説明します。

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

表 3-83 : ProcessingNode リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ		出力される個数	サポート構成	verbose が false の場合の表示
ProcessingNode		1	クラスタ	○
--	SystemConfiguration	1	クラスタ	×
	-- mode	1	クラスタ	×
	status	1	クラスタ	×
	name	1	クラスタ	×
	LU	1	クラスタ	×
	-- model	1	クラスタ	×
	serial	1	クラスタ	×
	ldev	1	クラスタ	×
	PhysicalNode	2	クラスタ	○
	-- location	1	クラスタ	×
	hostName	1	クラスタ	○
	physicalNodeName	1	クラスタ	×
	status	1	クラスタ	×
	managementLANAddress	1	クラスタ	×
	managementLANAddressIPv6	1	クラスタ	×
	HBLLanAddress	1	クラスタ	×
	BMCLanAddress	1	クラスタ	×
	resourceGroupStatus	1	クラスタ	×
	resourceGroupErrorInfo	1	クラスタ	×
	runningNode	1	クラスタ	×
	serial	1	クラスタ	×
	SystemVersion	1	クラスタ	×
	-- name	1	クラスタ	×
	version	1	クラスタ	×

(凡例) ○ : 表示されます。× : 表示されません。

ProcessingNode リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

表 3-84 : ProcessingNode リソースに GET メソッドをリクエストしたときのプロパティ

プロパティ		説明
ProcessingNode		--
--	SystemConfiguration	--
--	mode	構成状態が表示されます。 CLUSTER クラスタ構成のときに表示されます。
	status	クラスタの状態が表示されます。*1 ACTIVE 正常に稼働している場合に表示されます。 INACTIVE 停止している場合に表示されます。 UNKNOWN*2 状態が確認できない場合に表示されます。 DISABLE 障害によってフェールオーバー機能が無効になっている場合に表示されます。 UNDEF クラスタ未定義のときに表示されます。
	name	クラスタ名が 22 バイト以内で表示されます。 クラスタを構築していないとき、空文字列が表示されます。
	LU	--
--	model	共有 LU が存在するストレージシステムのモデルが表示されます。
	serial	共有 LU が存在するストレージシステムのシリアル番号が 16 バイト以内で表示されます。 クラスタを構築していないときは、空文字列が表示されます。
	ldev	共有 LU に割り当てた LU の LDEV 番号が 5 バイト以内で表示されます。 共有 LU に LDEV 番号が割り当てられていない場合、またはクラスタを構築していない場合は、空文字列が表示されます。
PhysicalNode		--
--	location	ユニット名 (node 0 または node 1) が表示されます。 クラスタを構築していないときの相手ノードの場合は、空文字列が表示されます。
	hostName	ホスト名が表示されます。 UNDEF クラスタを構築していないときの相手ノードの場合に表示されます。
	physicalNodeName	ノード名が 22 バイト以内で表示されます。 クラスタを構築していないとき、空文字列が表示されます。

プロパティ		説明
	status	<p>ノードの状態が表示されます。<sup>*1</sup></p> <p>UP</p> <p>正常に稼働している場合に表示されます。</p> <p>DOWN</p> <p>異常終了して停止している場合に表示されます。</p> <p>INACTIVE</p> <p>計画停止状態です。</p> <p>UNKNOWN<sup>*2</sup></p> <p>状態が確認できない場合に表示されます。</p> <p>UNDEF</p> <p>クラスタが未定義です。</p>
	managementLANAddress	<p>管理 LAN の IP アドレスが表示されます。</p> <p>未設定の場合、またはクラスタを構築していないときの相手ノードの場合は、空文字列が表示されます。</p>
	managementLanAddressIPv6	<p>管理 LAN の IPv6 アドレスが表示されます。</p> <p>未設定の場合、またはクラスタを構築していないときの相手ノードの場合は、空文字列が表示されます。</p>
	HBLanAddress	<p>ハートビート LAN の IP アドレスが表示されます。</p> <p>クラスタを構築していないとき、空文字列が表示されます。</p>
	BMCLanAddress	<p>BMC ポートの IP アドレスが表示されます。</p> <p>未設定の場合は、空文字列が表示されます。</p>
	resourceGroupStatus	<p>リソースグループの状態が表示されます。</p> <p>Online</p> <p>稼働中の場合に表示されます。</p> <p>Online Ready<sup>*3</sup></p> <p>クラスタが停止中のため、リソースグループを起動できません。クラスタの状態が DISABLE の場合にも表示されます。</p> <p>Online Pending</p> <p>開始処理中の場合に表示されます。</p> <p>Online Maintenance<sup>*3</sup></p> <p>監視機能が解除されています。</p> <p>Initializing<sup>*3</sup></p> <p>初期化処理中の場合に表示されます。</p> <p>Discovery (exclusivity)<sup>*3</sup></p> <p>稼働前のオンライン処理中の場合に表示されます。</p> <p>Offline</p> <p>計画停止状態です。クラスタの状態が DISABLE の場合にも表示されます。</p> <p>Offline Pending</p> <p>停止処理中の場合に表示されます。</p> <p>Internal Error<sup>*3</sup></p> <p>内部エラーを検出している場合に表示されます。</p> <p>UNDEF<sup>*3</sup></p> <p>クラスタが定義されていません。</p>

プロパティ			説明
	resourceGroupErrorInfo		<p>リソースグループのエラー状態が表示されます。</p> <p>No error エラーが発生していない場合に表示されます。</p> <p>Internal error - not recoverable<sup>*3</sup> 回復できない内部エラーが発生した場合に表示されます。</p> <p>Monitor activity unknown<sup>*3</sup> メンテナンスモードの開始または停止処理中にエラーが発生しました。</p> <p>No available nodes<sup>*3</sup> フェールオーバーの最後のノードでモニター異常を検出しました。</p> <p>Node not available(exclusivity)<sup>*3</sup> リソースグループのフェールオーバー先ノードが HA サービス停止中です。</p> <p>Node unknown<sup>*3</sup> ノードの状態が UNKNOWN のため、リソースグループが起動できません。</p> <p>Split resource group(exclusivity)<sup>*3</sup> クラスタ内でリソースグループが重複して稼働している場合に表示されます。</p> <p>srmd executable error<sup>*3</sup> 起動または停止処理中にエラーが発生している場合に表示されます。</p> <p>OS error 起動または停止処理中にエラーが発生している場合に表示されます。</p> <p>UNDEF<sup>*3</sup> クラスタが定義されていません。</p>
	runningNode		<p>リソースグループが稼働しているノード名が 22 バイト以内で表示されます。</p> <p>未設定の場合、空文字列が表示されます。</p>
	serial		<p>シリアル番号が 64 バイト以内で表示されます。</p> <p>クラスタを構築していない場合、または相手ノードと通信できないときの相手ノードの場合は、空文字列が表示されます。</p>
	SystemVersion		--
	--	name	プロダクト名が表示されます。
		version	OS のバージョンが 11 バイトで表示されます。

注 \*1 システムに障害が発生した場合、クラスタやノードの状態が表示されないことがあります。クラスタやノードの状態を確認できない場合、障害情報を収集してカスタマーサポートセンターに連絡してください。障害情報を収集する方法については、「トラブルシューティングガイド」(IF208A)を参照してください。

注 \*2 停止しているノードに対してメソッドを実行すると、クラスタおよびもう一方のノードの状態は、実際の状態に関わらず、UNKNOWN が表示されます。クラスタ、ノードおよびリソースグループの状態を確認するためには、稼働中のノード（もう一方のノード）に対して実行してください。また、クラスタ起動時は、クラスタを構成するノード上のすべての OS の起動が完了するまで（最大で 10 分程度）、UNKNOWN が表示されます。

注 \*3 クラスタ構成の場合だけ表示されます。

ProcessingNode リソースに GET メソッドをリクエストして、クラスタの情報を取得する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/ProcessingNode

#### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ProcessingNode>
  <PhysicalNode>
    <hostName>D6BQLNBX</hostName>
  </PhysicalNode>
</ProcessingNode>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-85 : ProcessingNode リソースに GET メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

## 3.11 Routing

### 3.11.1 Routing リソースの概要

Routing リソースを使用して、ルーティング情報を取得、追加、削除できます。

Routing リソースに使用できる HTTP メソッドを次の表に示します。

表 3-86 : Routing リソースに使用できる HTTP メソッド

リソースの URI	HTTP メソッド	サポート構成	説明
/Routings	GET	クラスタ Virtual Server	現在インターフェースに設定されているルーティングの情報を取得します。 クラスタ構成のノード上でメソッドを実行する場合は、Virtual Server に設定されているルーティングの情報も取得できます。Virtual Server の状態が Online/No error または Partial online/No error であることを確認してからメソッドを実行してください。
	POST	クラスタ Virtual Server	ルーティング情報を追加または削除します。

### 3.11.2 Routing リソース

この項では、Routing リソースの使い方について説明します。

#### 3.11.2.1 GET メソッドで指定したルーティング情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

表 3-87 : Routing リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ			出力される個数	サポート構成	verbose が false の場合の表示
Routings			1	クラスタ Virtual Server	○
--	Routing		0, 1 ~ n	クラスタ Virtual Server	○
	--	name	1	クラスタ Virtual Server	○
		type	1	クラスタ Virtual Server	○
		target	1	クラスタ Virtual Server	○
		netmask	1	クラスタ Virtual Server	○
		gateway	1	クラスタ Virtual Server	○
		isReject	1	クラスタ Virtual Server	○
		mss	1	クラスタ Virtual Server	×
	RoutingIPv6		0, 1 ~ n	クラスタ Virtual Server	○
	--	name	1	クラスタ Virtual Server	○
		type	1	クラスタ Virtual Server	○
		target	1	クラスタ Virtual Server	○
		prefixLength	1	クラスタ Virtual Server	○
		gateway	1	クラスタ Virtual Server	○
		isReject	1	クラスタ Virtual Server	○

(凡例) n : 2 以上の整数を示します。○ : 表示されます。× : 表示されません。

Routing リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

表 3-88 : Routing リソースに GET メソッドをリクエストしたときのプロパティ

プロパティ			説明
Routings			--
--	Routing		--
	--	name	インターフェースの名称が表示されます。



プロパティ			説明
--		type	ルーティングの宛先が表示されます。 net ネットワークが設定されている場合に表示されます。 host ホストが設定されている場合に表示されます。 default デフォルトルートが設定されている場合に表示されます。
		target	ルーティングの宛先がホスト名、IP アドレスまたはネットワークアドレスで表示されます。 デフォルトルートが設定されている場合は default と表示されます。
		netmask	ルーティングの宛先にネットワークが設定されている場合に、ネットマスクが表示されます。 ホストが設定されている場合は、空文字列が表示されます。 デフォルトルートが設定されている場合は 0.0.0.0 と表示されます。
		gateway	ゲートウェイのホスト名または IP アドレスが表示されます。 ゲートウェイが設定されていない場合は、空文字列が表示されます。
		isReject	宛先対象を拒否するかどうかが表示されます。 true 宛先対象を拒否する場合に表示されます。 false 宛先対象を拒否しない場合に表示されます。
		mss	経路の TCP 接続の最大セグメントサイズが 5 バイト以内で表示されます。 設定されていない場合は、空文字列が表示されます。なお、このとき最大セグメントサイズには、インターフェースに設定されている MTU 値から 40 を引いた値がストレージシステム内部で設定されています。
		RoutingIPv6	--
	--	name	インターフェースの名称が表示されます。
		type	ルーティングの宛先の設定方法が表示されます。 net ネットワークが設定されている場合に表示されます。 host ホストが設定されている場合に表示されます。 default デフォルトルートが設定されている場合に表示されます。
		target	ルーティングの宛先がホスト名、IPv6 アドレスまたはネットワークアドレスで表示されます。 デフォルトルートが設定されている場合は default と表示されます。
		prefixLength	ルーティングの宛先にネットワークが設定されている場合に、プレフィックス長が表示されます。 128 ホストが設定されている場合に表示されます。 default デフォルトルートが設定されている場合に表示されます。
		gateway	ゲートウェイのホスト名または IPv6 アドレスが表示されます。 ゲートウェイが設定されていない場合は、空文字列が表示されます。
		isReject	宛先対象を拒否するかどうかが表示されます。 true 宛先対象を拒否する場合に表示されます。 false 宛先対象を拒否しない場合に表示されます。

Routing リソースに GET メソッドをリクエストして、ルーティング情報を取得する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/Routings

### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Routings>
  <Routing>
    <name>mng0</name>
    <type>default</type>
    <target>default</target>
    <netmask>0.0.0.0</netmask>
    <gateway>192.168.0.1</gateway>
    <isReject>>false</isReject>
    <mss></mss>
    <dhcp>Off</dhcp>
  </Routing>
  <RoutingIPv6>
    <name>mng0</name>
    <type>net</type>
    <target>fdfl:afcb:44f9:1000::2:0</target>
    <prefixLength>64</prefixLength>
    <gateway>fdfl:afcb:44f9:1000::2:1</gateway>
    <isReject>>false</isReject>
  </RoutingIPv6>
</Routings>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-89： Routing リソースに GET メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3：」を参照してください。

## 3.11.2.2 POST メソッドでルーティング情報を追加または削除する

Routing リソースに POST メソッドをリクエストするときの XML の構造を次の表に示します。

表 3-90： Routing リソースに POST メソッドをリクエストするときの XML の構造

プロパティ		データ型	指定できる個数	サポート構成
Routing		--	1	クラスタ Virtual Server
--	operation	String	1	クラスタ Virtual Server
	name	String	1	クラスタ Virtual Server
	type	String	0, 1	クラスタ Virtual Server
	target	String	0, 1	クラスタ Virtual Server
	netmask	String	0, 1	クラスタ Virtual Server
	gateway	String	0, 1	クラスタ Virtual Server
	isReject	Boolean	0, 1	クラスタ Virtual Server
RoutingIPv6		--	1	クラスタ Virtual Server

プロパティ		データ型	指定できる個数	サポート構成
--	operation	String	1	クラスタ Virtual Server
	name	String	1	クラスタ Virtual Server
	type	String	0, 1	クラスタ Virtual Server
	target	String	0, 1	クラスタ Virtual Server
	prefixLength	String	0, 1	クラスタ Virtual Server
	gateway	String	0, 1	クラスタ Virtual Server
	isReject	Boolean	0, 1	クラスタ Virtual Server

Routing リソースに POST メソッドをリクエストするときに指定するプロパティについて次の表に示します。

**表 3-91 : Routing リソースに POST メソッドをリクエストするときのプロパティ**

プロパティ		説明	指定時の条件
Routing		--	--
--	operation	ルーティング情報に対する操作方法を設定します。 Add ルーティング情報を追加する場合に指定します。 Delete ルーティング情報を削除する場合に指定します。	必須
	name	ルーティング情報を追加するインターフェースの名称を設定します。 • eth <番号> • agr <番号> • rdn <番号> • xgbe <番号>	必須
	type <sup>*1</sup>	ルーティングの宛先を指定する方法を設定します。 net ネットワークをルーティングの宛先に設定する場合に指定します。 host ホストをルーティングの宛先に設定する場合に指定します。 default デフォルトルートを設定する場合に指定します。	operation プロパティが Add の場合に必須です。
	target <sup>*2</sup>	ルーティングの宛先のホスト名、IP アドレスまたはネットワークアドレスを設定します。	次のどちらかの場合に必須です。 • operation プロパティが Add、かつ type プロパティが net または host である • operation プロパティが Delete である

プロパティ		説明	指定時の条件
	netmask <sup>*1</sup>	ネットマスクを設定します。	type プロパティが net の場合は必須です。
	gateway	ゲートウェイのホスト名または IP アドレスを設定します。 省略した場合、値は設定されません。	operation プロパティが Add の場合は、次の条件をすべて満たすときに必須です。 <ul style="list-style-type: none"> <li>type プロパティが default である</li> <li>isReject プロパティが false である</li> </ul> operation プロパティが Delete の場合は、ゲートウェイが設定されているときに必須です。
	isReject	設定するルーティング情報に対して、経路を拒否するかどうかを設定します。 true 経路を拒否する場合に指定します。 false <sup>*3</sup> 経路を設定する場合に指定します。	次の条件をすべて満たす場合は必ず true を指定します。 <ul style="list-style-type: none"> <li>type プロパティが default である</li> <li>gateway プロパティを指定しない</li> </ul> operation プロパティが Add の場合に指定できます。
RoutingIPv6		--	--
--	operation	ルーティング情報に対する操作方法を設定します。 Add ルーティング情報を追加する場合に指定します。 Delete ルーティング情報を削除する場合に指定します。	必須
	name	ルーティング情報を追加するインターフェースの名称を設定します。 <ul style="list-style-type: none"> <li>eth &lt;番号&gt;</li> <li>agr &lt;番号&gt;</li> <li>rdn &lt;番号&gt;</li> <li>xgbe &lt;番号&gt;</li> </ul>	必須
	type	ルーティングの宛先を指定する方法を設定します。 net ネットワークをルーティングの宛先に設定する場合に指定します。 host ホストをルーティングの宛先に設定する場合に指定します。 default デフォルトルートを設定する場合に指定します。	operation プロパティが Add の場合に必須です。
	target	ルーティングの宛先のホスト名、IPv6 アドレスまたはネットワークアドレスを設定します。	次のどちらかの場合に必須です。 <ul style="list-style-type: none"> <li>operation プロパティが Add、かつ type プロパティが net または host である</li> <li>operation プロパティが Delete である</li> </ul>

プロパティ		説明	指定時の条件
	prefixLength	プレフィックス長を設定します。 128 ホストをルーティングの宛先に設定する場合に指定します。 default デフォルトルートを設定する場合に指定します。	type プロパティが net の場合は必須です。
	gateway	ゲートウェイのホスト名または IPv6 アドレスを設定します。 省略した場合、値は設定されません。	operation プロパティが Add の場合は、次の条件をすべて満たすときに必須です。 <ul style="list-style-type: none"> <li>type プロパティが default である</li> <li>isReject プロパティが false である</li> </ul> operation プロパティが Delete の場合は、ゲートウェイが設定されているときに必須です。
	isReject	設定するルーティング情報に対して、経路を拒否するかどうかを設定します。 true 経路を拒否する場合に指定します。 false <sup>*3</sup> 経路を設定する場合に指定します。	次の条件をすべて満たす場合は必ず true を指定します。 <ul style="list-style-type: none"> <li>type プロパティが default である</li> <li>gateway プロパティを指定しない</li> </ul> operation プロパティが Add の場合に指定できます。

注意：

宛先が同じルーティング情報を複数追加しないでください。ルーティング情報を正しく削除できなくなるおそれがあります。

ゲートウェイの設定以外がすべて同じルーティング情報は設定しないでください。ルーティング情報を正しく削除できなくなるおそれがあります。

注 \*1

- ・ type プロパティで net を選択し、netmask プロパティに「0.0.0.0」を指定して追加したルーティング情報は、デフォルトルートとして動作します。
- ・ type プロパティで net を選択し、netmask プロパティに「255.255.255.255」を指定して追加したルーティング情報は、ルーティングの宛先にホストを直接指定したものとして動作します。

注 \*2

ルーティングの宛先には、次のネットワークの IP アドレスは指定できません。これらの IP アドレスはストレージシステム内部で予約されています。

IPv4 の場合：

127.0.0.0 ～ 127.255.255.255

IPv6 の場合：

::ffff: < IPv4 アドレス >, :: < IPv4 アドレス >, ::1/128, ::/0, ::/ 128, fe80::/10, ff00::/8

- ・ 保守ポートに設定されている IP アドレスのネットワーク
- ・ ハートビートポートに設定されている IP アドレスのネットワーク

保守ポートに設定されている IP アドレスは pmctl コマンドで確認できます。

ハートビートポートに設定されている IP アドレスは、ProcessingNode リソースを使用して確認してください。ProcessingNode リソースについては、「[3.10.2 ProcessingNode リソース](#)」を参照してください。

注 \*3

プロパティを省略した場合のデフォルト値です。

Routing リソースに POST メソッドをリクエストして、ルーティング情報を追加する例を示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/Routings

#### リクエスト XML の例 (IPv4 で設定する場合)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Routing>
  <operation>Add</operation>
  <name>eth1</name>
  <type>host</type>
  <target>nec.japan.com</target>
  <netmask>255.255.255.0</netmask>
  <gateway>155.98.31.1</gateway>
</Routing>
```

#### リクエスト XML の例 (IPv6 で設定する場合)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<RoutingIPv6>
  <operation>Add</operation>
  <name>eth1</name>
  <type>net</type>
  <prefixLength>64</prefixLength>
  <target>3ffe:0200:0000:010a:0000:0000:0000:0000</target>
  <gateway>3ffe:0200:0000:010a:0000:0000:0000:0001</gateway>
</RoutingIPv6>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-92 : Routing リソースに POST メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

## 3.12 SystemController

### 3.12.1 SystemController リソースの概要

SystemController リソースを使用して、サービスと OS を制御できます。

SystemController リソースに使用できる HTTP メソッドを次の表に示します。

表 3-93 : SystemController リソースに使用できる HTTP メソッド

リソースの URI	HTTP メソッド	サポート構成	説明
/SystemController	GET	クラスタ Virtual Server	サービスの再起動が必要かどうかを確認します。
	POST	クラスタ Virtual Server	サービスを再起動します。

リソースの URI	HTTP メソッド	サポート構成	説明
/SystemController/OS	GET	クラスタ Virtual Server	OS の起動状態を確認します。
	POST	クラスタ	ノード上の OS を停止します。

## 3.12.2 SystemController リソース

この項では、SystemController リソースの使い方について説明します。

### 3.12.2.1 GET メソッドでサービスまたは OS の再起動が必要かどうかを確認する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

**表 3-94 : SystemController リソースに GET メソッドをリクエストしたときのレスポンス XML の構造**

プロパティ			出力される個数	サポート構成	verbose が false の場合の表示
Services			1	クラスタ Virtual Server	○
--	Service		1 ~ n	クラスタ Virtual Server	○
	--	serviceName	1	クラスタ Virtual Server	○
		isRestart	1	クラスタ Virtual Server	○

(凡例) n : 2 以上の整数を示します。○ : 表示されます。

SystemController リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

**表 3-95 : SystemController リソースに GET メソッドをリクエストしたときのプロパティ**

プロパティ			説明
Services			--
--	Service		--
	--	serviceName	サービス名が表示されます。 CIFS CIFS サービスです。 NFS NFS サービスです。 FTP FTP サービスです。 SFTP SFTP サービスです。 TFTP TFTP サービスです。
		isRestart	サービスの再起動が必要かどうかが表示されます。 true 再起動が必要な場合に表示されます。 false 再起動が必要ない場合に表示されます。

SystemController リソースに GET メソッドをリクエストして、サービスおよび OS の再起動が必要かどうかを確認する例を次に示します。

### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/SystemController

### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Services>
  <Service>
    <serviceName>CIFS</serviceName>
    <isRestart>true</isRestart>
  </Service>
  <Service>
    <serviceName>NFS</serviceName>
    <isRestart>true</isRestart>
  </Service>
  <Service>
    <serviceName>FTP</serviceName>
    <isRestart>true</isRestart>
  </Service>
  <Service>
    <serviceName>OS</serviceName>
    <isRestart>true</isRestart>
  </Service>
  <Service>
    <serviceName>SFTP</serviceName>
    <isRestart>true</isRestart>
  </Service>
  <Service>
    <serviceName>TFTP</serviceName>
    <isRestart>true</isRestart>
  </Service>
</Services>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

表 3-96 : SystemController リソースに GET メソッドをリクエストしたときのステータスコード

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.12.2.2 POST メソッドでサービスまたは OS を再起動する

SystemController リソースに POST メソッドをリクエストするときの XML の構造を次の表に示します。

表 3-97 : SystemController リソースに POST メソッドをリクエストするときの XML の構造

プロパティ			データ型	指定できる個数	サポート構成
Services			--	1	クラスタ Virtual Server
--	service		--	1 ~ n	クラスタ Virtual Server
	--	serviceName	String	1	クラスタ Virtual Server
		isRestart	String	1	クラスタ Virtual Server

(凡例) n : 2 以上の整数を示します。

SystemController リソースに POST メソッドをリクエストするときに指定するプロパティについて次の表に示します。



表 3-98 : SystemController リソースに POST メソッドをリクエストするときのプロパティ

プロパティ		説明	指定時の条件
Services		--	--
--	service	--	--
	serviceName	サービス名を設定します。 CIFS CIFS サービスを制御する場合に指定します。 NFS NFS サービスを制御する場合に指定します。 FTP FTP サービスを制御する場合に指定します。 SFTP SFTP サービスを制御する場合に指定します。 TFTP TFTP サービスを制御する場合に指定します。	必須
	isRestart	サービスを再起動するかどうかを設定します。 true サービスまたは OS を再起動する場合に指定します。 false サービスまたは OS を再起動しない場合に指定します。	必須

SystemController リソースに POST メソッドをリクエストして、サービスおよび OS を再起動する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/SystemController

#### リクエスト XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Services>
  <Service>
    <serviceName>CIFS</serviceName>
    <isRestart>true</isRestart>
  </Service>
  <Service>
    <serviceName>NFS</serviceName>
    <isRestart>true</isRestart>
  </Service>
  <Service>
    <serviceName>OS</serviceName>
    <isRestart>true</isRestart>
  </Service>
  <Service>
    <serviceName>FTP</serviceName>
    <isRestart>true</isRestart>
  </Service>
  <Service>
    <serviceName>SFTP</serviceName>
    <isRestart>true</isRestart>
  </Service>
  <Service>
    <serviceName>TFTP</serviceName>
    <isRestart>true</isRestart>
  </Service>
</Services>
```

</Services>

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-99 : SystemController リソースに POST メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
202	メソッドを受け付けました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

## 3.12.3 SystemController/OS リソース

この項では、SystemController/OS リソースの使い方について説明します。

### 3.12.3.1 GET メソッドで OS の起動状態を確認する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

**表 3-100 : SystemController/OS リソースに GET メソッドをリクエストしたときのレスポンス XML の構造**

プロパティ		出力される個数	サポート構成	verbose が false の場合の表示
Service		1	クラスタ Virtual Server	○
--	status	1	クラスタ Virtual Server	×

(凡例) ○ : 表示されます。× : 表示されません。

SystemController/OS リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

**表 3-101 : SystemController/OS リソースに GET メソッドをリクエストしたときのプロパティ**

プロパティ		説明
Service		--
--	status	OS の起動状態が表示されます。 Starting OS が起動処理中の場合に表示されます。 Running OS の起動処理が完了している場合に表示されます。

SystemController/OS リソースに GET メソッドをリクエストして、OS の起動状態を確認する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/SystemController/OS

#### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Service>
  <status>Running</operation>
</Service>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-102 : SystemController/OS リソースに GET メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

### 3.12.3.2 POST メソッドでノード上の OS を再起動または停止する

注意 :

リクエストは、実行したノードだけで有効になります。クラスタ構成の場合でノード上の OS を停止するときは、両方のノードに対して同じ POST メソッドを同時にリクエストする必要があります。両方のノードに同時にリクエストしないと、フェールオーバーが発生するおそれがあります。

SystemController/OS リソースに POST メソッドをリクエストするときの XML の構造を次の表に示します。

**表 3-103 : SystemController/OS リソースに POST メソッドをリクエストするときの XML の構造**

プロパティ	データ型	指定できる個数	サポート構成
Service	--	1	クラスタ
--	operation	String	1
			クラスタ

SystemController/OS リソースに POST メソッドをリクエストするときに指定するプロパティについて次の表に示します。

**表 3-104 : SystemController/OS リソースに POST メソッドをリクエストするときのプロパティ**

プロパティ	説明	指定時の条件
Service	--	--
--	operation  Shutdown  OS を停止します。クラスタ構成の場合は、両方のノードで実行してください。	必須

SystemController/OS リソースに POST メソッドをリクエストして、ノード上の OS を停止する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/SystemController/OS

#### リクエスト XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Service>
  <operation>Shutdown</operation>
</Service>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-105 : SystemController/OS リソースに POST メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
202	メソッドを受け付けました。
その他	異常終了しました。「表 1-3 : 」を参照してください。

## 3.13 VirtualServer

### 3.13.1 VirtualServer リソースの概要

VirtualServer リソースを使用して、ログインしている Virtual Server の情報を取得できます。

VirtualServer リソースに使用できる HTTP メソッドを次の表に示します。

表 3-106 : VirtualServer リソースに使用できる HTTP メソッド

リソースの URI	HTTP メソッド	サポート構成	説明
/VirtualServer	GET	Virtual Server	ログインしている Virtual Server の情報を取得します。

### 3.13.2 VirtualServer リソース

この項では、VirtualServer リソースの使い方について説明します。

#### 3.13.2.1 GET メソッドで Virtual Server の情報を取得する

メソッドが終了したときに出力されるレスポンス XML の構造を次の表に示します。

表 3-107 : VirtualServer リソースに GET メソッドをリクエストしたときのレスポンス XML の構造

プロパティ		出力される個数	サポート構成	verbose が false の場合の表示
VirtualServer		1	Virtual Server	○
--	id	1	Virtual Server	○
	Name	1	Virtual Server	○
	serviceStatus	1	Virtual Server	○
	errorStatus	1	Virtual Server	○

(凡例) ○ : 表示されます。

VirtualServer リソースに GET メソッドをリクエストしたときに表示されるプロパティについて次の表に示します。

表 3-108 : VirtualServer リソースに GET メソッドをリクエストしたときのプロパティ

プロパティ		説明
VirtualServer		--
--	id	Virtual Server ID が 3 バイト以内で表示されます。
	Name	Virtual Server 名が 15 バイト以内で表示されます。

プロパティ		説明
	serviceStatus	<p>Virtual Server の状態が表示されます。</p> <p>Online pending 起動処理中の場合に表示されます。</p> <p>Online 稼働中の場合に表示されます。</p> <p>Partial online 稼働中ですが、一部のサービスが停止している場合に表示されます。</p> <p>Offline pending 停止処理中の場合に表示されます。</p> <p>Error エラーが発生している場合に表示されます。エラー情報を参照して対処してください。</p>
	errorStatus	<p>Virtual Server のエラー情報が表示されます。</p> <p>No error エラーが発生していない場合に表示されます。</p> <p>Internal error 回復できない内部エラーが発生している場合に表示されます。</p> <p>OS error Virtual Server が起動、停止、再起動またはフェールオーバーできない場合に表示されます。</p>

VirtualServer リソースに GET メソッドをリクエストして、Virtual Server の情報を取得する例を次に示します。

#### リクエスト先の URI

https://<ホスト名またはIPアドレス>:9090/mapi/VirtualServer

#### レスポンス XML の例

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<VirtualServer>
  <id>111</id>
  <name>vserver1</name>
  <serviceStatus>online</serviceStatus>
  <errorStatus>Internal error</errorStatus>
</VirtualServer>
```

メソッドが終了したときに出力されるステータスコードを次の表に示します。

**表 3-109 : VirtualServer リソースに GET メソッドをリクエストしたときのステータスコード**

ステータスコード	説明
200	メソッドの実行が正常終了しました。
その他	異常終了しました。「表 1-3 : 」を参照してください。



# API の使用例

ここでは、HVFP の API を使用するための例を示します。

- ❑ [A.1 クラスタおよびノードの情報を取得する例](#)
- ❑ [A.2 ハードウェア状態を監視する例](#)
- ❑ [A.3 ネットワークを設定する例](#)
- ❑ [A.4 ファイルシステムを作成する例](#)
- ❑ [A.5 ファイルシステムを削除する例](#)
- ❑ [A.6 OS を停止する例](#)

## A.1 クラスタおよびノードの情報を取得する例

クラスタおよびノードの情報を取得する例を次に示します。

表 A-1: クラスタおよびノードの情報を取得する例

操作	リソースの URI	HTTP メソッド	参照先
1. クラスタおよびノードの情報を取得します。	/ProcessingNode	GET	3.10.2.1

1. クラスタおよびノードの情報を取得します。

コマンドライン:

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または IP アドレス>:9090/mapi/ProcessingNode?prettyprint&verbose=true" -k -X GET -H "Accept: application/xml"
```

レスポンス:

```
HTTP/1.1 200 OK
Date: Tue, 12 Apr 2011 05:18:42 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 1759
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 2
X-NODE-OperationList: [1]Get node details, [2]Get system details
Connection: close
Content-Type: application/xml
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ProcessingNode>
  <SystemConfiguration>
    <mode>CLUSTER</mode>
  </SystemConfiguration>
  <status>ACTIVE</status>
  <name>cluster</name>
  <LU>
    <serial>83016761</serial>
    <ldev>0</ldev>
  </LU>
  <PhysicalNode>
    <location>node 0</location>
    <hostName>DT019000037</hostName>
    <physicalNodeName>DT019000037</physicalNodeName>
    <status>UP</status>
    <managementLANAddress>10.213.88.250</managementLANAddress>
    <HBLanAddress>172.23.212.21</HBLanAddress>
    <BMCLanAddress>10.213.88.249</BMCLanAddress>
    <resourceGroupStatus>Online</resourceGroupStatus>
    <resourceGroupErrorInfo>No error</resourceGroupErrorInfo>
    <runningNode>DT019000037</runningNode>
    <serial>4230MB0-T019000037</serial>
    <SystemVersion>
      <name>Virtual File Platform</name>
      <version>5.5.1-00</version>
    </SystemVersion>
  </PhysicalNode>
  <PhysicalNode>
    <location>node 1</location>
    <hostName>DT019000042</hostName>
    <physicalNodeName>DT019000042</physicalNodeName>
    <status>UP</status>
    <managementLANAddress>10.213.88.251</managementLANAddress>
    <HBLanAddress>172.23.212.25</HBLanAddress>
    <BMCLanAddress>10.213.88.248</BMCLanAddress>
    <resourceGroupStatus>Online</resourceGroupStatus>
    <resourceGroupErrorInfo>No error</resourceGroupErrorInfo>
    <runningNode>DT019000042</runningNode>
    <serial>4230MB0-T019000042</serial>
    <SystemVersion>
      <name></name>
      <version></version>
    </SystemVersion>
  </PhysicalNode>
</ProcessingNode>
```



## A.2 ハードウェア状態を監視する例

ハードウェア状態を監視する例を次に示します。

表 A-2: ハードウェア状態を監視する例

操作	リソースの URI	HTTP メソッド	参照先
1. ハードウェア情報を取得します。	/Hardware	GET	3.6.2.1
2. FC バスの情報を取得します。	/FCPaths	GET	3.4.2.1

1. ハードウェア情報を取得します。

コマンドライン:

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または IP アドレス>:9090/mapi/Hardware?prettyprint" -k -X GET -H "Accept: application/xml"
```

レスポンス:

```
HTTP/1.1 200 OK
Date: Tue, 12 Apr 2011 05:30:27 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 7994
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get hardware details
Connection: close
Content-Type: application/xml
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Hardware>
  <PortDetails>
    <PortDetail>
      <status>Up</status>
      <portName>eth1</portName>
      <mediaType>copper</mediaType>
      <linkSpeed>1000Base</linkSpeed>
    </PortDetail>
    <PortDetail>
      <status>Up</status>
      <portName>eth2</portName>
      <mediaType>copper</mediaType>
      <linkSpeed>1000Base</linkSpeed>
    </PortDetail>
    <PortDetail>
      <status>Down</status>
      <portName>eth3</portName>
      <mediaType>copper</mediaType>
      <linkSpeed>Unknown!</linkSpeed>
    </PortDetail>
  </PortDetails>
  <ManagementLans>
    <ManagementLan>
      <status>Up</status>
      <portName>mng0</portName>
      <mediaType>copper</mediaType>
      <linkSpeed>100Base</linkSpeed>
    </ManagementLan>
  </ManagementLans>
  <HeartbeatLans>
    <HeartbeatLan>
      <status>Up</status>
      <portName>hb0</portName>
      <mediaType>copper</mediaType>
      <linkSpeed>100Base</linkSpeed>
    </HeartbeatLan>
  </HeartbeatLans>
  <PrivateMaintenanceLans>
    <PrivateMaintenanceLan>
      <status>Up</status>
      <portName>pm0</portName>
      <mediaType>copper</mediaType>
      <linkSpeed>100Base</linkSpeed>
    </PrivateMaintenanceLan>
    <PrivateMaintenanceLan>
      <status>Down</status>
    </PrivateMaintenanceLan>
  </PrivateMaintenanceLans>
</Hardware>
```

```

        <portName>pml</portName>
        <mediaType>copper</mediaType>
        <linkSpeed>Unknown!</linkSpeed>
    </PrivateMaintenanceLan>
</PrivateMaintenanceLans>
<InternalHDDs>
    <InternalHDD>
        <slotName>0</slotName>
        <status>Normal</status>
        <vendorInformation>size:136.732 GB,raid level:RAID1</
vendorInformation>
    </InternalHDD>
    <InternalHDD>
        <slotName>1</slotName>
        <status>Normal</status>
        <vendorInformation>size:136.732 GB,raid level:RAID1</
vendorInformation>
    </InternalHDD>
</InternalHDDs>
<Memories>
    <Memory>
        <slotName>0</slotName>
        <status>Not Installed</status>
        <vendorInformation>locator:P1C0D1,size:-</vendorInformation>
    </Memory>
    <Memory>
        <slotName>1</slotName>
        <status>Installed</status>
        <vendorInformation>locator:P1C0D0,size:2048 MB</
vendorInformation>
    </Memory>
    <Memory>
        <slotName>2</slotName>
        <status>Not Installed</status>
        <vendorInformation>locator:P1C1D1,size:-</vendorInformation>
    </Memory>
    <Memory>
        <slotName>3</slotName>
        <status>Installed</status>
        <vendorInformation>locator:P1C1D0,size:2048 MB</
vendorInformation>
    </Memory>
    <Memory>
        <slotName>4</slotName>
        <status>Not Installed</status>
        <vendorInformation>locator:P1C2D1,size:-</vendorInformation>
    </Memory>
    <Memory>
        <slotName>5</slotName>
        <status>Installed</status>
        <vendorInformation>locator:P1C2D0,size:2048 MB</
vendorInformation>
    </Memory>
</Memories>
<MemTotals>
    <MemTotal>
        <slotName>0</slotName>
        <size>size:11.5 GB(12.0 GB)</size>
    </MemTotal>
</MemTotals>
<PowerSupplies>
    <PowerSupply>
        <powerSupplyName>0</powerSupplyName>
        <status>OK</status>
        <vendorInformation></vendorInformation>
    </PowerSupply>
    <PowerSupply>
        <powerSupplyName>1</powerSupplyName>
        <status>OK</status>
        <vendorInformation></vendorInformation>
    </PowerSupply>
</PowerSupplies>
<Fans>
    <Fan>
        <fanName>0</fanName>
        <status>OK</status>
        <vendorInformation></vendorInformation>
    </Fan>
</Fans>

```

```

        <fanName>1</fanName>
        <status>OK</status>
        <vendorInformation></vendorInformation>
    </Fan>
    <Fan>
        <fanName>2</fanName>
        <status>OK</status>
        <vendorInformation></vendorInformation>
    </Fan>
    <Fan>
        <fanName>3</fanName>
        <status>OK</status>
        <vendorInformation></vendorInformation>
    </Fan>
</Fans>
<Temperatures>
    <Temperature>
        <temperatureName>0</temperatureName>
        <status>OK</status>
        <vendorInformation>reading:55 (+/- 0.500) degrees C</
vendorInformation>
    </Temperature>
    <Temperature>
        <temperatureName>1</temperatureName>
        <status>OK</status>
        <vendorInformation>reading:44 (+/- 0.500) degrees C</
vendorInformation>
    </Temperature>
    <Temperature>
        <temperatureName>2</temperatureName>
        <status>OK</status>
        <vendorInformation>reading:42 (+/- 0.500) degrees C</
vendorInformation>
    </Temperature>
</Temperatures>
<BMCs>
    <BMC>
        <status>OK</status>
        <connection>OK</connection>
    </BMC>
</BMCs>
<Models>
    <Model>
        <manufacturer>NEC</manufacturer>
        <product>Express5800/R120d-2M [N8100-1963Y]</product>
    </Model>
</Models>
<Serials>
    <Serial>
        <number>4230MB0-T019000037</number>
    </Serial>
</Serials>
</Hardware>

```

## 2. FCパスの情報を取得します。

コマンドライン：

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または
IPアドレス>:9090/mapi/FCPaths?prettyprint" -k -X GET -H "Accept:
application/xml"
```

レスポンス：

```
HTTP/1.1 200 OK
Date: Tue, 12 Apr 2011 02:43:12 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 809
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get FC path details
Connection: close
Content-Type: application/xml
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FCPaths>
    <FCPath>
        <path>path000-0004-1E</path>
        <target>N1-T000</target>
        <hostPort>fc0004</hostPort>
        <hostPortWWN>10000000c98f2992</hostPortWWN>
        <arrayPort>1E</arrayPort>
    </FCPath>
</FCPaths>

```

```

        <arrayPortWWN>50060e801046889c</arrayPortWWN>
        <serial>83016761</serial>
        <status>Online</status>
    </FCPath>
    <FCPath>
        <path>path000-0005-1F</path>
        <target>N1-T000</target>
        <hostPort>fc0005</hostPort>
        <hostPortWWN>10000000c98f2993</hostPortWWN>
        <arrayPort>1F</arrayPort>
        <arrayPortWWN>50060e801046889d</arrayPortWWN>
        <serial>83016761</serial>
        <status>Online</status>
    </FCPath>
</FCPaths>

```

## A.3 ネットワークを設定する例

ネットワークを設定する例を次に示します。

表 A-3：ネットワークを設定する例

操作	リソースの URI	HTTP メソッド	参照先
1. ネットワークインターフェースの情報を取得します。	/NetworkInterfaces	GET	3.8.2.1
2. ネットワークインターフェースを追加します。	/NetworkInterfaces	PUT	3.8.2.2
3. 追加したネットワークインターフェースを確認します。	/NetworkInterfaces/ <ネットワークインターフェース名>	HEAD	3.8.3.3
4. 指定したネットワークインターフェースの情報を取得します。	/NetworkInterfaces/ <ネットワークインターフェース名>	GET	3.8.3.1
5. 指定したネットワークインターフェースの情報を変更します。	/NetworkInterfaces/ <ネットワークインターフェース名>	POST	3.8.3.2
6. ルーティング情報を取得します。	/Routings	GET	3.11.2.1
7. ルーティング情報を設定します。	/Routings	POST	3.11.2.2

1. ネットワークインターフェースの情報を取得します。

コマンドライン：

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または IP アドレス>:9090/mapi/NetworkInterfaces?prettyprint&verbose=true" -k -X GET -H "Accept: application/xml"
レスポンス：

```

```

HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 13:24:21 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 3398
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get network interface details
Connection: close
Content-Type: application/xml

```

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NetworkInterfaces>
  <NetworkInterface>
    <name>eth1</name>
    <mtu></mtu>
    <netmask></netmask>
    <networkAddress></networkAddress>
    <FixedIP>
      <target></target>
      <other></other>
    </FixedIP>
  </NetworkInterface>
</NetworkInterfaces>

```

```

        <ServiceIP>
            <target></target>
            <other></other>
        </ServiceIP>
    </NetworkInterface>
    <NetworkInterface>
        <name>eth2</name>
        <mtu></mtu>
        <netmask></netmask>
        <networkAddress></networkAddress>
        <FixedIP>
            <target></target>
            <other></other>
        </FixedIP>
        <ServiceIP>
            <target></target>
            <other></other>
        </ServiceIP>
    </NetworkInterface>
    <NetworkInterface>
        <name>eth3</name>
        <mtu></mtu>
        <netmask></netmask>
        <networkAddress></networkAddress>
        <FixedIP>
            <target></target>
            <other></other>
        </FixedIP>
        <ServiceIP>
            <target></target>
            <other></other>
        </ServiceIP>
    </NetworkInterface>
    <NetworkInterface>
        <name>mng0</name>
        <mtu>1500</mtu>
        <netmask>255.255.252.0</netmask>
        <networkAddress>10.213.88.0</networkAddress>
        <FixedIP>
            <target>10.213.88.250</target>
            <other>10.213.88.251</other>
        </FixedIP>
        <ServiceIP>
            <target>10.213.88.252</target>
            <other>10.213.88.253</other>
        </ServiceIP>
    </NetworkInterface>
</NetworkInterfaces>

```

## 2. ネットワークインターフェースを追加します。

```

ifset.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NetworkInterface>
    <name>eth1</name>
    <netmask>255.255.255.0</netmask>
    <FixedIP>
        <target>192.168.30.10</target>
        <other>192.168.30.11</other>
    </FixedIP>
    <ServiceIP>
        <target>192.168.30.12</target>
        <other>192.168.30.13</other>
    </ServiceIP>
</NetworkInterface>

```

コマンドライン：

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/NetworkInterfaces" -k -d @ifset.xml -X PUT -H
"Content-Type: application/xml"

```

レスポンス：

```

HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 13:24:23 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Set network interface
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1

```

3. 追加したネットワークインターフェースを確認します。

コマンドライン :

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/NetworkInterfaces/eth1" -k -I -X HEAD -H "Content-Type: application/xml"
```

レスポンス :

```
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 13:24:35 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 300
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get network interface details
Connection: close
Content-Type: application/xml
```

4. 指定したネットワークインターフェースの情報を取得します。

コマンドライン :

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/NetworkInterfaces/eth1?prettyprint&verbose=true" -k -X GET -H "Accept: application/xml"
```

レスポンス :

```
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 13:24:39 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 439
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get network interface details
Connection: close
Content-Type: application/xml
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NetworkInterface>
  <name>eth1</name>
  <mtu>1500</mtu>
  <netmask>255.255.255.0</netmask>
  <networkAddress>192.168.30.0</networkAddress>
  <FixedIP>
    <target>192.168.30.10</target>
    <other>192.168.30.11</other>
  </FixedIP>
  <ServiceIP>
    <target>192.168.30.12</target>
    <other>192.168.30.13</other>
  </ServiceIP>
</NetworkInterface>
```

5. 指定したネットワークインターフェースの情報を変更します。

ifmod.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NetworkInterface>
  <netmask>255.255.255.0</netmask>
  <FixedIP>
    <target>192.168.30.14</target>
    <other>192.168.30.15</other>
  </FixedIP>
  <ServiceIP>
    <target>192.168.30.16</target>
    <other>192.168.30.17</other>
  </ServiceIP>
</NetworkInterface>
```

コマンドライン :

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>/mapi/NetworkInterfaces/eth1" -k -d @ifmod.xml -X POST -H "Content-Type: application/xml"
```

レスポンス :

```
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 13:24:42 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Edit network interface
```

```
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1
```

6. ルーティング情報を取得します。

コマンドライン：

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/Routings?prettyprint&verbose=true" -k -H
"Content-type: application/xml"
レスポンス：
```

```
HTTP/1.1 200 OK
Date: Fri, 09 Sep 2011 23:11:04 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 555
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get routing details
Connection: close
Content-Type: application/xml
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Routings>
  <Routing>
    <name>mng0</name>
    <type>default</type>
    <target>default</target>
    <netmask>0.0.0.0</netmask>
    <gateway>192.168.0.1</gateway>
    <isReject>>false</isReject>
  </Routing>
  <RoutingIPv6>
    <name>eth1</name>
    <type>net</type>
    <target>3ffe:200:0:10a::0</target>
    <prefixLength>64</prefixLength>
    <gateway>3ffe:200:0:10a::1</gateway>
    <isReject>>false</isReject>
  </RoutingIPv6>
</Routings>
```

7. ルーティング情報を設定します。

```
rtsetv6.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<RoutingIPv6>
  <operation>Add</operation>
  <name>eth1</name>
  <type>net</type>
  <prefixLength>64</prefixLength>
  <target>3ffe:0200:0000:010a:0000:0000:0000:0000</target>
  <gateway>3ffe:0200:0000:010a:0000:0000:0000:0001</gateway>
</RoutingIPv6>
```

コマンドライン：

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/Routings" -d @rtsetv6.xml -X POST -k -H "Content-
Type: application/xml"
レスポンス：
```

```
HTTP/1.1 200 OK
Date: Fri, 09 Sep 2011 23:10:51 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Add routing details
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1
```

## A.4 ファイルシステムを作成する例

ファイルシステムを作成する例を次に示します。

表 A-4: ファイルシステムを作成する例

操作	リソースの URI	HTTP メソッド	参照先
1. LU の情報を取得します。	/LUs	GET	3.7.2.1
2. ファイルシステムの情報を取得します。	/FileSystems	GET	3.5.2.1
3. ファイルシステムを作成します。	/FileSystems	PUT	3.5.2.2
4. 作成したファイルシステムのマウント設定の情報を取得します。	/FileSystems/ <ファイルシステム名> / MountSetting	GET	3.5.4.1
5. 作成したファイルシステムをマウントします。	/FileSystems/ <ファイルシステム名> / MountSetting	POST	3.5.4.2
6. 編集するファイルシステムの情報を取得します。	/FileSystems/ <ファイルシステム名>	GET	3.5.3.1
7. 作成したファイルシステムの情報を編集します。	/FileSystems/ <ファイルシステム名>	POST	3.5.3.2
8. CIFS 共有の情報を取得します。	/CIFSShares	GET	3.3.2.1
9. CIFS 共有を作成します。	/CIFSShares	PUT	3.3.2.2
10. 編集する CIFS 共有の情報を取得します。	/CIFSShares/ < Base64 でエンコードされた CIFS 共有名>	GET	3.3.3.1
11. CIFS 共有の情報を編集します。	/CIFSShares/ < Base64 でエンコードされた CIFS 共有名>	POST	3.3.3.2
12. NFS 共有の情報を取得します。	/NFSShares	GET	3.9.2.1
13. NFS 共有を作成します。	/NFSShares	PUT	3.9.2.2
14. 編集する NFS 共有の情報を取得します。	/NFSShares/ < NFS 共有ディレクトリ名>	GET	3.9.3.1
15. NFS 共有の情報を編集します。	/NFSShares/ < NFS 共有ディレクトリ名>	POST	3.9.3.2

## 1. LU の情報を取得します。

ファイルシステムに使用する LU を確認します。

コマンドライン:

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または IP アドレス>:9090/mapi/LUs?prettyprint" -k -X GET -H "Accept: application/xml"
```

レスポンス:

```
HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 07:56:29 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 19272
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get LU details
Connection: close
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<LUs>
  <LU>
    <deviceFileName>lu0000</deviceFileName>
    <userLUN>0000</userLUN>
    <isExternalDevice>>false</isExternalDevice>
    <target>N1-T000</target>
    <serial>83016761</serial>
```



```

        <ldev>1</ldev>
        <type>SATA</type>
        <size>20.000</size>
        <function>Free</function>
        <usedBy></usedBy>
        <rg>000</rg>
        <dp></dp>
        <usedPage></usedPage>
        <pool></pool>
        <freePool></freePool>
        <totalPool></totalPool>
    </LU>
    <LU>
        <deviceFileName>lu0001</deviceFileName>
        <userLUN>0001</userLUN>
        <isExternalDevice>false</isExternalDevice>
        <target>N1-T000</target>
        <serial>83016761</serial>
        <ldev>2</ldev>
        <type>SATA</type>
        <size>20.000</size>
        <function>Free</function>
        <usedBy></usedBy>
        <rg>000</rg>
        <dp></dp>
        <usedPage></usedPage>
        <pool></pool>
        <freePool></freePool>
        <totalPool></totalPool>
    </LU>
    <LU>
        <deviceFileName>lu0004</deviceFileName>
        <userLUN>0004</userLUN>
        <isExternalDevice>false</isExternalDevice>
        <target>N1-T000</target>
        <serial>83016761</serial>
        <ldev>5</ldev>
        <type>SATA</type>
        <size>20.000</size>
        <function>File</function>
        <usedBy>BCD</usedBy>
        <rg>000</rg>
        <dp></dp>
        <usedPage></usedPage>
        <pool></pool>
        <freePool></freePool>
        <totalPool></totalPool>
    </LU>
</LUs>

```

## 2. ファイルシステムの情報を取得します。

作成するファイルシステムのファイルシステム名などを決めるため、作成されているファイルシステムを確認します。

コマンドライン：

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems?prettyprint&verbose=true" -k -X GET -H "Accept: application/xml"
```

レスポンス：

```

HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 07:56:38 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 4399
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get file system details
Connection: close
Content-Type: application/xml

```

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FileSystems>
  <FileSystem>
    <fileName>fs01</fileName>
    <isLvmEnabled>true</isLvmEnabled>
    <isLvmStripingEnabled>false</isLvmStripingEnabled>
    <stripeSize></stripeSize>
  
```

```

    <aclType>Advanced ACL</aclType>
    <NamespaceShareSettings>
    <type>--</type>
    <FQDN></FQDN>
    <user></user>
    <password></password>
    <hcpReplicaHost></hcpReplicaHost>
    <hcpNamespace></hcpNamespace>
    </NamespaceShareSettings>
    <UseVersioning>not used</UseVersioning>
    <periodToHold></periodToHold>
    <isBypassEnabled>true</isBypassEnabled>
    <WormSetting>
    <maxRetention></maxRetention>
    <minRetention></minRetention>
    <isAutoCommitEnabled></isAutoCommitEnabled>
    <commitModeSetting></commitModeSetting>
    <autoCommitPeriod></autoCommitPeriod>
    <defaultRetention></defaultRetention>
    </WormSetting>
    <LUs>
    <LU>
    <deviceFileName>lu0001</deviceFileName>
    </LU>
    </LUs>
  </FileSystem>
  <FileSystem>
    <fileName>fs02</fileName>
    <isLvmEnabled>true</isLvmEnabled>
    <isLvmStripingEnabled>false</isLvmStripingEnabled>
    <stripeSize></stripeSize>
    <aclType>Classic ACL</aclType>
    <NamespaceShareSettings>
    <type>--</type>
    <FQDN></FQDN>
    <user></user>
    <password></password>
    <hcpReplicaHost></hcpReplicaHost>
    <hcpNamespace></hcpNamespace>
    </NamespaceShareSettings>
    <UseVersioning>not used</UseVersioning>
    <periodToHold></periodToHold>
    <isBypassEnabled>true</isBypassEnabled>
    <WormSetting>
    <maxRetention></maxRetention>
    <minRetention></minRetention>
    <isAutoCommitEnabled></isAutoCommitEnabled>
    <commitModeSetting></commitModeSetting>
    <autoCommitPeriod></autoCommitPeriod>
    <defaultRetention></defaultRetention>
    </WormSetting>
    <LUs>
    <LU>
    <deviceFileName>lu0002</deviceFileName>
    </LU>
    </LUs>
  </FileSystem>
  <FileSystem>
    <fileName>fs05</fileName>
    <isLvmEnabled>true</isLvmEnabled>
    <isLvmStripingEnabled>false</isLvmStripingEnabled>
    <stripeSize></stripeSize>
    <aclType>Classic ACL</aclType>
    <NamespaceShareSettings>
    <type>Read/Write</type>
    <FQDN></FQDN>
    <user></user>
    <password></password>
    <hcpReplicaHost></hcpReplicaHost>
    <hcpNamespace>fs05-6d84d799-1b40-427c-85a6-53e6b75d3e3e</hcpNamespace>
    </NamespaceShareSettings>
    <UseVersioning>used</UseVersioning>
    <periodToHold>7</periodToHold>
    <isBypassEnabled>true</isBypassEnabled>
    <WormSetting>
    <maxRetention>36500-0-0</maxRetention>
    <minRetention>0-0-0</minRetention>
    <isAutoCommitEnabled>true</isAutoCommitEnabled>

```

```

        <commitModeSetting>manual</commitModeSetting>
        <autoCommitPeriod>0-0-15</autoCommitPeriod>
        <defaultRetention>10950-0-0</defaultRetention>
    </WormSetting>
    <LUs>
    <LU>
        <deviceFileName>lu0003</deviceFileName>
    </LU>
    <LU>
        <deviceFileName>lu0005</deviceFileName>
    </LU>
    </LUs>
</FileSystem>
</FileSystems>

```

### 3. ファイルシステムを作成します。

fscreate.xml

```

<FileSystem>
    <fileName>fs</fileName>
    <isLvmEnabled>true</isLvmEnabled>
    <isLvmStripingEnabled>false</isLvmStripingEnabled>
    <stripeSize></stripeSize>
    <aclType>Classic ACL</aclType>
    <NamespaceShareSettings>
    <type>Read/Write</type>
    </NamespaceShareSettings>
    <UseVersioning>used</UseVersioning>
    <periodToHold>7</periodToHold>
    <isBypassEnabled>true</isBypassEnabled>
    <WormSetting>
    <maxRetention>36500-0-0</maxRetention>
    <minRetention>0-0-0</minRetention>
    <isAutoCommitEnabled>true</isAutoCommitEnabled>
    <commitModeSetting>manual</commitModeSetting>
    <autoCommitPeriod>0-0-15</autoCommitPeriod>
    <defaultRetention>10950-0-0</defaultRetention>
    </WormSetting>
    <LUs>
    <LU>
        <deviceFileName>lu0003</deviceFileName>
    </LU>
    <LU>
        <deviceFileName>lu0005</deviceFileName>
    </LU>
    </LUs>
</FileSystem>

```

コマンドライン：

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems" -k -d @fscreate.xml -X PUT -H
"Content-Type: application/xml"

```

レスポンス：

```

HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 07:56:42 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Create file system
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1

```

### 4. 作成したファイルシステムのマウント設定の情報を取得します。

ファイルシステムがマウントされていないことを確認します。

コマンドライン：

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems/fs/
MountSetting?prettyprint&verbose=true" -k -X GET -H "Accept:
application/xml"

```

レスポンス：

```

HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 07:56:58 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 264
X-NODE-APIVersionID: 5.5.1-00

```

```

X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get file system details
Connection: close
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<MountSetting>
  <isMounted>false</isMounted>
  <mountStatus></mountStatus>
  <isReadOnly></isReadOnly>
  <isFileLastAccessTimeRecordingEnabled></
isFileLastAccessTimeRecordingEnabled>
</MountSetting>

```

##### 5. 作成したファイルシステムをマウントします。

次の条件でマウントする例です。

- 書き込みを許可する
- ファイルシステムにアクセスしたときの最終アクセス時刻を更新する

```

fsmount.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<MountSetting>
  <isMounted>true</isMounted>
  <isReadOnly>false</isReadOnly>
  <isFileLastAccessTimeRecordingEnabled>true</
isFileLastAccessTimeRecordingEnabled>
</MountSetting>

```

コマンドライン:

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または
IPアドレス>:9090/mapi/FileSystems/fs/MountSetting" -k -d @fsmount.xml -
X POST -H "Content-Type: application/xml"
レスポンス:

```

```

HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 07:57:01 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Mount file system
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1

```

##### 6. 編集するファイルシステムの情報を取得します。

編集するファイルシステムに設定されている内容を確認するために、情報を取得します。

コマンドライン:

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または
IPアドレス>:9090/mapi/FileSystems/fs/?prettyprint&verbose=true" -k -X
GET -H "Accept: application/xml"
レスポンス:

```

```

HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 07:57:17 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 1424
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get file system details
Connection: close
Content-Type: application/xml

```

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FileSystem>
  <fileName>fs</fileName>
  <usage>WORM</usage>
  <deviceStatus>Normal</deviceStatus>
  <freeBlocksInMegaBytes>40870.912</freeBlocksInMegaBytes>
  <usedBlocksInMegaBytes>4.096</usedBlocksInMegaBytes>
  <totalBlocksInMegaBytes>40875.008</totalBlocksInMegaBytes>
  <isLvmEnabled>true</isLvmEnabled>
  <fileSystemSizeInGigaByte>40.000</fileSystemSizeInGigaByte>
  <isLvmStripingEnabled>true</isLvmStripingEnabled>
  <stripes>2</stripes>
  <stripeSize>64</stripeSize>
  <aclType>Advanced ACL</aclType>
  <WormSetting>

```

```

        <maxRetention>10950-0-0</maxRetention>
        <minRetention>0-0-0</minRetention>
        <isAutoCommitEnabled>false</isAutoCommitEnabled>
        <commitModeSetting></commitModeSetting>
        <autoCommitPeriod></autoCommitPeriod>
        <defaultRetention></defaultRetention>
        <isWormDeleteEnabled>true</isWormDeleteEnabled>
        <longestRetention></longestRetention>
    </WormSetting>
    <MountSetting>
        <isMounted>true</isMounted>
        <mountStatus>Normal</mountStatus>
        <isReadOnly>false</isReadOnly>
        <isFileLastAccessTimeRecordingEnabled>true</
isFileLastAccessTimeRecordingEnabled>
    </MountSetting>
    <LUs>
        <LU>
            <deviceFileName>lu0004</deviceFileName>
        </LU>
        <LU>
            <deviceFileName>lu0005</deviceFileName>
        </LU>
    </LUs>
</FileSystem>

```

## 7. 作成したファイルシステムの情報を編集します。

次の条件でファイルシステムの情報を編集する例です。

- **WORM** 対応ファイルシステムで自動コミットを有効にする
- ファイルシステムを拡張するために **LU** を追加する

```

fsmod.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FileSystem>
    <WormSetting>
        <isAutoCommitEnabled>true</isAutoCommitEnabled>
        <commitModeSetting>manual</commitModeSetting>
        <autoCommitPeriod>0-0-15</autoCommitPeriod>
        <defaultRetention>10950-0-0</defaultRetention>
    </WormSetting>
    <addLUs>
        <LU>
            <deviceFileName>lu0006</deviceFileName>
        </LU>
        <LU>
            <deviceFileName>lu0007</deviceFileName>
        </LU>
    </addLUs>
</FileSystem>

```

コマンドライン：

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または
IPアドレス>:9090/mapi/FileSystems/fs/" -k -d @fsmod.xml -X POST -H
"Content-Type: application/xml"
レスポンス：

```

```

HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 07:57:20 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 2
X-NODE-OperationList: [1]Expand file system, [2]Edit file system
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1

```

## 8. CIFS 共有の情報を取得します。

作成する **CIFS** 共有の共有名などを決めるため、作成されている **CIFS** 共有を確認します。

コマンドライン：

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または
IPアドレス>:9090/mapi/CIFSShares?prettyprint&verbose=true" -k -X GET -H
"Accept: application/xml"
レスポンス：

```

```

HTTP/1.1 200 OK
Date: Thu, 21 Feb 2013 06:42:36 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk

```

```

Content-Length: 1519
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get CIFS share details
Connection: close
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CIFSShares>
  <CIFSShare>
    <exportPoint>/mnt/aaa/sh1</exportPoint>
    <cifsShareName>sh</cifsShareName>
    <isAclEnabled>true</isAclEnabled>
    <restrictionTargetHosts>0.0.0/0</restrictionTargetHosts>
    <isAccessRestrictionAllowed>>false</isAccessRestrictionAllowed>
    <commentForShare></commentForShare>
    <isReadOnly>>false</isReadOnly>
    <isBrowseEnabled>true</isBrowseEnabled>
    <fileAccessPermissionOfOwner></fileAccessPermissionOfOwner>
    <fileAccessPermissionOfGroup></fileAccessPermissionOfGroup>
    <fileAccessPermissionOfOther></fileAccessPermissionOfOther>
    <directoryAccessPermissionOfOwner></
directoryAccessPermissionOfOwner>
    <directoryAccessPermissionOfGroup></
directoryAccessPermissionOfGroup>
    <directoryAccessPermissionOfOther></
directoryAccessPermissionOfOther>
    <guestAccessPermission>default</guestAccessPermission>
    <diskSynchronizationPolicy>default</diskSynchronizationPolicy>
    <cifsClientCacheSetting>default</cifsClientCacheSetting>
    <fileTimestampChangeableUsers></fileTimestampChangeableUsers>
    <isHomeDirectoryEnabled>>false</isHomeDirectoryEnabled>
    <clientAccessPolicy>default</clientAccessPolicy>
    <vssUse>default</vssUse>
    <cifsClientRoCacheOptimize>default</cifsClientRoCacheOptimize>
    <accessBasedEnumeration>default</accessBasedEnumeration>
    <writeDisallowedUsers></writeDisallowedUsers>
    <writeDisallowedGroups></writeDisallowedGroups>
    <writeAllowedUsers></writeAllowedUsers>
    <writeAllowedGroups></writeAllowedGroups>
  </CIFSShare>
</CIFSShares>

```

## 9. CIFS 共有を作成します。

CIFS 共有を次の条件で作成する例です。

- 共有ディレクトリとして、/mnt/fs/cifs ディレクトリを新たに作成する
- CIFS 共有名は cifsshare とする
- ファイルシステムにアクセスしたときの最終アクセス時刻を更新する
- アクセスを制限しない
- CIFS クライアントに ACL の参照・設定を許可する
- 書き込みを許可する
- CIFS 共有でホームディレクトリの自動作成機能を有効にする
- CIFS 共有への書き込みを許可および拒否するユーザー、グループを設定する
- ACL を設定する

```

cifscshare.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CIFSShare>
  <exportPoint>/mnt/fs/cifs</exportPoint>
  <cifsShareName>cifsshare</cifsShareName>
  <isClassicAcl>>false</isClassicAcl>
  <isAclEnabled>true</isAclEnabled>
  <restrictionTargetHosts>10.213.77.231</restrictionTargetHosts>
  <isAccessRestrictionAllowed>true</isAccessRestrictionAllowed>
  <commentForShare>cifscomment</commentForShare>
  <isReadOnly>>false</isReadOnly>
  <isBrowseEnabled>true</isBrowseEnabled>
  <guestAccessPermission>default</guestAccessPermission>
  <diskSynchronizationPolicy>default</diskSynchronizationPolicy>
  <cifsClientCacheSetting>default</cifsClientCacheSetting>
  <isHomeDirectoryEnabled>true</isHomeDirectoryEnabled>
  <clientAccessPolicy>default</clientAccessPolicy>
  <vssUse>default</vssUse>
  <cifsClientRoCacheOptimize>default</cifsClientRoCacheOptimize>

```

```

<accessBasedEnumeration>default</accessBasedEnumeration>
<writeDisallowedUsers>user3,user4</writeDisallowedUsers>
<writeDisallowedGroups>group3,group4</writeDisallowedGroups>
<writeAllowedUsers>user1,user2</writeAllowedUsers>
<writeAllowedGroups>group1,group2</writeAllowedGroups>
<DirectorySetup>
  <userName>22222</userName>
  <groupName>11111</groupName>
  <isStickyBit>true</isStickyBit>
  <ownerPermission>rw</ownerPermission>
  <groupPermission>rw</groupPermission>
  <otherPermission>rw</otherPermission>
</DirectorySetup>
<DirectoryACLSettings>
  <DirectoryACLSetting>
    <userGroupAccess>allow</userGroupAccess>
    <name>Everyone</name>
    <accountType>g</accountType>
    <applicationDestination>a</applicationDestination>
    <inheritanceRange>a</inheritanceRange>
    <accessPermission>r</accessPermission>
  </DirectoryACLSetting>
</DirectoryACLSettings>
</CIFSShare>

```

コマンドライン：

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または
IPアドレス>:9090/mapi/CIFSShares" -k -d @cifscreate.xml -X PUT -H
"Content-Type: application/xml"
レスポンス：
HTTP/1.1 100 Continue

HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 07:57:36 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 4
X-NODE-OperationList: [1]Create shared directory, [2]Edit shared
directory, [3]Set ACL, [4]Create CIFS share
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1

```

## 10.編集する CIFS 共有の情報を取得します。

編集する CIFS 共有に設定されている内容を確認するために、情報を取得します。

コマンドライン：

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または
IPアドレス>:9090/mapi/CIFSShares/Y2lmc3NoYXJl?SHARE-DIRECTORY-
NAME=L2ludC9mcy9jaWZz&prettyprint&verbose=true" -k -X GET -H "Accept:
application/xml"
レスポンス：
HTTP/1.1 200 OK
Date: Thu, 21 Feb 2013 06:29:58 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 1519

X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get CIFS share details
Connection: close
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CIFSShare>
  <exportPoint>/mnt/fs/cifs</exportPoint>
  <cifsShareName>cifsshare</cifsShareName>
  <isAclEnabled>true</isAclEnabled>
  <restrictionTargetHosts>0.0.0.0/0</restrictionTargetHosts>
  <isAccessRestrictionAllowed>>false</isAccessRestrictionAllowed>
  <commentForShare></commentForShare>
  <isReadOnly>>false</isReadOnly>
  <isBrowseEnabled>true</isBrowseEnabled>
  <fileAccessPermissionOfOwner></fileAccessPermissionOfOwner>
  <fileAccessPermissionOfGroup></fileAccessPermissionOfGroup>

```

```

        <fileAccessPermissionOfOther></fileAccessPermissionOfOther>
        <directoryAccessPermissionOfOwner></
directoryAccessPermissionOfOwner>
        <directoryAccessPermissionOfGroup></
directoryAccessPermissionOfGroup>
        <directoryAccessPermissionOfOther></
directoryAccessPermissionOfOther>
        <guestAccessPermission>default</guestAccessPermission>
        <diskSynchronizationPolicy>default</diskSynchronizationPolicy>
        <cifsClientCacheSetting>default</cifsClientCacheSetting>
        <fileTimestampChangeableUsers></fileTimestampChangeableUsers>
        <isHomeDirectoryEnabled>false</isHomeDirectoryEnabled>
        <clientAccessPolicy>default</clientAccessPolicy>
        <vssUse>default</vssUse>
        <cifsClientRoCacheOptimize>default</cifsClientRoCacheOptimize>
        <accessBasedEnumeration>default</accessBasedEnumeration>
        <writeDisallowedUsers></writeDisallowedUsers>
        <writeDisallowedGroups></writeDisallowedGroups>
        <writeAllowedUsers></writeAllowedUsers>
        <writeAllowedGroups></writeAllowedGroups>
</CIFSShare>

```

## 11. CIFS 共有の情報を編集します。

次の条件で CIFS 共有の情報を編集する例です。

- CIFS 共有名を変更する
- CIFS 共有への書き込みを許可および拒否するユーザー、グループを変更する

```

cifsmod.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CIFSShare>
  <cifsShareName>cifssharechange</cifsShareName>
  <isClassicAcl>false</isClassicAcl>
  <isAclEnabled>true</isAclEnabled>
  <restrictionTargetHosts>10.213.77.88</restrictionTargetHosts>
  <isAccessRestrictionAllowed>true</isAccessRestrictionAllowed>
  <commentForShare>cifscomment</commentForShare>
  <isReadOnly>false</isReadOnly>
  <isBrowseEnabled>true</isBrowseEnabled>
  <guestAccessPermission>default</guestAccessPermission>
  <diskSynchronizationPolicy>default</diskSynchronizationPolicy>
  <cifsClientCacheSetting>default</cifsClientCacheSetting>
  <isHomeDirectoryEnabled>true</isHomeDirectoryEnabled>
  <clientAccessPolicy>default</clientAccessPolicy>
  <vssUse>default</vssUse>
  <cifsClientRoCacheOptimize>default</cifsClientRoCacheOptimize>
  <accessBasedEnumeration>default</accessBasedEnumeration>
  <writeDisallowedUsers>user1,user2</writeDisallowedUsers>
  <writeDisallowedGroups>group1,group2</writeDisallowedGroups>
  <writeAllowedUsers>user3,user4</writeAllowedUsers>
  <writeAllowedGroups>group3,group4</writeAllowedGroups>
</CIFSShare>

```

コマンドライン：

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または
IPアドレス>:9090/mapi/CIFSShares/Y2lmc3NoYXJl?SHARE-DIRECTORY-
NAME=L2ludC9mcy9jaWZz" -k -X POST -d @cifsmod.xml -H "Content-Type:
application/xml"

```

レスポンス：

```
HTTP/1.1 100 Continue
```

```
HTTP/1.1 200 OK
```

```
Date: Tue, 19 Apr 2011 07:57:52 GMT
```

```
Server: Nec Web Server 04-00 (Unix) mod_jk
```

```
X-NODE-APIVersionID: 5.5.1-00
```

```
X-NODE-CompletedOperation: 1
```

```
X-NODE-OperationList: [1]Edit CIFS share
```

```
Content-Length: 0
```

```
Connection: close
```

```
Content-Type: text/plain; charset=iso-8859-1
```

## 12. NFS 共有の情報を取得します。

作成する NFS 共有の共有名などを決めるため、作成されている NFS 共有を確認します。

コマンドライン：

```

curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または
IPアドレス>:9090/mapi/NFSShares?prettyprint&verbose=true" -k -X GET -H
"Accept: application/xml"

```



レスポンス :

```
HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 07:57:56 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 925
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get NFS share details
Connection: close
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NFSShares>
  <NFSShare>
    <exportPoint>/mnt/aaa/add</exportPoint>
    <anonymousUID>65534</anonymousUID>
    <anonymousGID>65534</anonymousGID>
    <isPortRestrictionEnabled>false</isPortRestrictionEnabled>
    <isSubtreeCheckEnabled>false</isSubtreeCheckEnabled>
    <isAccessCheckWithLockRequestEnabled>false</
isAccessCheckWithLockRequestEnabled>
    <aclType>Classic ACL</aclType>
    <maxRwSize></maxRwSize>
    <AllowedHosts>
      <AllowedHost>
        <hostOrAddress>*</hostOrAddress>
        <isReadOnly>false</isReadOnly>
        <synchronousWritingSetting>rw_sync</
synchronousWritingSetting>
        <anonymousMapping>root_only</anonymousMapping>
        <securityFlavor>sys</securityFlavor>
      </AllowedHost>
    </AllowedHosts>
  </NFSShare>
</NFSShares>
```

### 13.NFS 共有を作成します。

NFS 共有を次の条件で作成する例です。

- 共有ディレクトリとして、/mnt/fs/nfs ディレクトリを新たに作成する
- 匿名ユーザーとしてアクセスするユーザー ID、グループ ID を設定する
- 1024 番より小さなインターネットポートから発信されたリクエストだけを受け付ける
- NFS 共有に対する最大転送長を 5KB とする
- すべてのホストに NFS 共有を読み取り専用で公開する
- ACL を設定する

```
nfscreate.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NFSShare>
  <exportPoint>/mnt/fs/nfs</exportPoint>
  <anonymousUID>22222</anonymousUID>
  <anonymousGID>22222</anonymousGID>
  <isPortRestrictionEnabled>true</isPortRestrictionEnabled>
  <isSubtreeCheckEnabled>true</isSubtreeCheckEnabled>
  <isAccessCheckWithLockRequestEnabled>true</
isAccessCheckWithLockRequestEnabled>
  <maxRwSize>5</maxRwSize>
  <AllowedHosts>
    <AllowedHost>
      <hostOrAddress>*</hostOrAddress>
      <isReadOnly>true</isReadOnly>
      <synchronousWritingSetting>rw_sync</
synchronousWritingSetting>
      <anonymousMapping>root_only</anonymousMapping>
      <securityFlavor>sys</securityFlavor>
    </AllowedHost>
  </AllowedHosts>
  <DirectorySetup>
    <userName>22222</userName>
    <groupName>11111</groupName>
    <isStickyBit>true</isStickyBit>
    <ownerPermission>rw</ownerPermission>
    <groupPermission>rw</groupPermission>
    <otherPermission>rw</otherPermission>
  </DirectorySetup>
  <DirectoryACLSettings>
```

```

        <DirectoryACLSetting>
          <userGroupAccess>allow</userGroupAccess>
          <name>Everyone</name>
          <accountType>g</accountType>
          <applicationDestination>a</applicationDestination>
          <inheritanceRange>a</inheritanceRange>
          <accessPermission>r</accessPermission>
        </DirectoryACLSetting>
      </DirectoryACLSettings>
    </NFSShare>
  コマンドライン :
  curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または
  IPアドレス>:9090/mapi/NFSShares" -k -d @nfscshare.xml -X PUT -H
  "Content-Type: application/xml"

  レスポンス :
  HTTP/1.1 200 OK
  Date: Tue, 19 Apr 2011 07:57:58 GMT
  Server: Nec Web Server 04-00 (Unix) mod_jk
  X-NODE-APIVersionID: 5.5.1-00
  X-NODE-CompletedOperation: 4
  X-NODE-OperationList: [1]Create shared directory, [2]Edit shared
  directory, [3]Set ACL, [4]Create NFS share
  Content-Length: 0
  Connection: close
  Content-Type: text/plain; charset=iso-8859-1

```

#### 14. 編集する NFS 共有の情報を取得します。

編集する NFS 共有に設定されている内容を確認するために、情報を取得します。

```

  コマンドライン :
  curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または
  IPアドレス>:9090/mapi/NFSShares/nfs?SHARE-DIRECTORY-NAME=/mnt/fs/
  nfs&prettyprint&verbose=true" -k -X GET -H "Accept: application/xml"
  レスポンス :
  HTTP/1.1 200 OK
  Date: Tue, 19 Apr 2011 07:58:16 GMT
  Server: Nec Web Server 04-00 (Unix) mod_jk
  Content-Length: 821
  X-NODE-APIVersionID: 5.5.1-00
  X-NODE-CompletedOperation: 1
  X-NODE-OperationList: [1]Get NFS share details
  Connection: close
  Content-Type: application/xml

  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
  <NFSShare>
    <exportPoint>/mnt/fs/nfs</exportPoint>
    <anonymousUID>22222</anonymousUID>
    <anonymousGID>22222</anonymousGID>
    <isPortRestrictionEnabled>true</isPortRestrictionEnabled>
    <isSubtreeCheckEnabled>true</isSubtreeCheckEnabled>
    <isAccessCheckWithLockRequestEnabled>true</
  isAccessCheckWithLockRequestEnabled>
    <aclType>Advanced ACL</aclType>
    <maxRwSize>5</maxRwSize>
    <AllowedHosts>
      <AllowedHost>
        <hostOrAddress>*</hostOrAddress>
        <isReadOnly>true</isReadOnly>
        <synchronousWritingSetting>rw_sync</
  synchronousWritingSetting>
        <anonymousMapping>root_only</anonymousMapping>
        <securityFlavor>sys</securityFlavor>
      </AllowedHost>
    </AllowedHosts>
  </NFSShare>

```

#### 15. NFS 共有の情報を編集します。

次の条件で NFS 共有の情報を編集する例です。

- 匿名ユーザーとしてアクセスするユーザー ID, グループ ID を設定する
- すべてのインターネットポートから発信されたリクエストを受け付ける
- サブツリーに実行権限 (x) があるかどうかはチェックしない

- ロック要求があった場合に、権限チェックをしない
- NFS 共有に対する最大転送長を 6KB とする
- すべてのホストに NFS 共有への書き込みを許可する

```
nfsmod.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NFSShare>
  <anonymousUID>33333</anonymousUID>
  <anonymousGID>33333</anonymousGID>
  <isPortRestrictionEnabled>>false</isPortRestrictionEnabled>
  <isSubtreeCheckEnabled>>false</isSubtreeCheckEnabled>
  <isAccessCheckWithLockRequestEnabled>>false</
isAccessCheckWithLockRequestEnabled>
  <maxRwSize>6</maxRwSize>
  <AllowedHosts>
    <AllowedHost>
      <hostOrAddress>*</hostOrAddress>
      <isReadOnly>>false</isReadOnly>
      <synchronousWritingSetting>rw_sync</
synchronousWritingSetting>
      <anonymousMapping>root_only</anonymousMapping>
      <securityFlavor>sys</securityFlavor>
    </AllowedHost>
  </AllowedHosts>
</NFSShare>
```

コマンドライン：

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または
IPアドレス>:9090/mapi/NFSShares/nfs?SHARE-DIRECTORY-NAME=/mnt/fs/nfs" -
k -d @nfsmod.xml -X POST -H "Content-Type: application/xml"
```

```
レスポンス：
HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 07:58:17 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Edit NFS share
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1
```

## A.5 ファイルシステムを削除する例

ファイルシステムを削除する例を次に示します。

表 A-5：ファイルシステムを削除する例

操作	リソースの URI	HTTP メソッド	参照先
1. 指定した CIFS 共有があるかどうか確認します。	/CIFSShares/ <Base64 でエンコードされた CIFS 共有名>	HEAD	<a href="#">3.3.3.4</a>
2. 指定した CIFS 共有を削除します。	/CIFSShares/ <Base64 でエンコードされた CIFS 共有名>	DELETE	<a href="#">3.3.3.3</a>
3. 指定した NFS 共有があるかどうか確認します。	/NFSShares/ < NFS 共有ディレクトリ名>	HEAD	<a href="#">3.9.3.4</a>
4. 指定した NFS 共有を削除します。	/NFSShares/ < NFS 共有ディレクトリ名>	DELETE	<a href="#">3.9.3.3</a>
5. 指定したファイルシステムがあるかどうか確認します。	/FileSystems/ <ファイルシステム名>	HEAD	<a href="#">3.5.3.4</a>
6. 指定したファイルシステムをアンマウントします。	/FileSystems/ <ファイルシステム名> /MountSetting	POST	<a href="#">3.5.3.2</a>

操作	リソースの URI	HTTP メソッド	参照先
7. 指定したファイルシステムを削除します。	/FileSystems/ <ファイルシステム名>	DELETE	3.5.3.3

1. 指定した CIFS 共有があるかどうか確認します。

コマンドライン:

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または IP アドレス>:9090/mapi/CIFSShares/Y2lmc3NoYXJlY2hhbmdl?SHARE-DIRECTORY-NAME=L2ludC9mcy9jaWZz" -k -X HEAD -H "Accept: application/xml"
```

レスポンス:

```
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 05:39:02 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 1499
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get CIFS share details
Connection: close
Content-Type: application/xml
```

2. 指定した CIFS 共有を削除します。

コマンドライン:

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または IP アドレス>:9090/mapi/CIFSShares/Y2lmc3NoYXJlY2hhbmdl?SHARE-DIRECTORY-NAME=L2ludC9mcy9jaWZz" -k -X DELETE -H "Accept: application/xml"
```

レスポンス:

```
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 05:39:03 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Delete CIFS share
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1
```

3. 指定した NFS 共有があるかどうか確認します。

コマンドライン:

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または IP アドレス>:9090/mapi/NFSShares/nfs?SHARE-DIRECTORY-NAME=/mnt/fs/nfs" -k -X HEAD -H "Accept: application/xml"
```

レスポンス:

```
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 05:39:05 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 658
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get NFS share details
Connection: close
Content-Type: application/xml
```

4. 指定した NFS 共有を削除します。

コマンドライン:

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名または IP アドレス>:9090/mapi/NFSShares/nfs?SHARE-DIRECTORY-NAME=/mnt/fs/nfs" -k -X DELETE -H "Accept: application/xml"
```

レスポンス:

```
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 05:39:07 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Delete NFS share
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1
```

5. 指定したファイルシステムがあるかどうか確認します。

コマンドライン：

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems/fs/" -k -X HEAD -H "Content-Type: application/xml"
レスポンス：
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 05:39:11 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
Content-Length: 685
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Get file system details
Connection: close
Content-Type: application/xml
```

6. 指定したファイルシステムをアンマウントします。

```
fsumount.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<MountSetting>
  <isMounted>false</isMounted>
  <isReadOnly>false</isReadOnly>
  <isFileLastAccessTimeRecordingEnabled>true</
isFileLastAccessTimeRecordingEnabled>
</MountSetting>
コマンドライン：
```

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems/fs/MountSetting" -k -d @fsumount.xml -X POST -H "Content-Type: application/xml"
レスポンス：
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 05:39:14 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Unmount file system
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1
```

7. 指定したファイルシステムを削除します。

コマンドライン：

```
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<ホスト名またはIPアドレス>:9090/mapi/FileSystems/fs/" -k -X DELETE -H "Content-Type: application/xml"
レスポンス：
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2011 05:39:27 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1>Delete file system
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=iso-8859-1
```

## A.6 OS を停止する例

OS を停止する例を次に示します。

表 A-6：OS を停止する例

操作	リソースの URI	HTTP メソッド	参照先
1. リクエスト先のノードの OS を停止します。	/SystemController/OS	POST	3.12.3.2
2. クラスタ内のもう一方のノードの OS を停止します。	/SystemController/OS	POST	3.12.3.2

1. リクエスト先のノードの OS を停止します。

```
osShutdown.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Service>
  <operation>Shutdown</operation>
</Service>
コマンドライン :
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<片方のノードの
ホスト名>:9090/mapi/SystemController/OS -k -d @osShutdown.xml -X POST -
H Content-Type: application/xml
レスポンス :
HTTP/1.1 202 Accepted
Date: Tue, 12 Apr 2011 13:21:18 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Shutdown OS
Connection: close
Transfer-Encoding: chunked
Content-Type: text/html; charset=utf-8

<html><head><title>202 Accepted</title></head><body><h1>202 Accepted</
h1></body></html>
```

2. クラスタ内のもう一方のノードの OS を停止します。

```
osShutdown.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Service>
  <operation>Shutdown</operation>
</Service>
コマンドライン :
curl -i -l -b api-auth=<ユーザー名>:<パスワード> "https://<もう一方のノードの
ホスト名>:9090/mapi/SystemController/OS -k -d @osShutdown.xml -X
POST -H Content-Type: application/xml
レスポンス :
HTTP/1.1 202 Accepted
Date: Tue, 12 Apr 2011 13:21:19 GMT
Server: Nec Web Server 04-00 (Unix) mod_jk
X-NODE-APIVersionID: 5.5.1-00
X-NODE-CompletedOperation: 1
X-NODE-OperationList: [1]Shutdown OS
Connection: close
Transfer-Encoding: chunked
Content-Type: text/html; charset=utf-8

<html><head><title>202 Accepted</title></head><body><h1>202 Accepted</
h1></body></html>
```

iStorage M シリーズ  
NAS オプション ソフトウェア  
Virtual File Platform  
**API リファレンス**

I F 2 1 2 A - 8

2 0 1 5 年 3 月 初 版  
2 0 1 8 年 3 月 8 版

日本電気株式会社

東京都港区芝五丁目 7 番 1 号

TEL(03)3454-1111 (大代表)

©NEC Corporation 2015, 2016, 2017, 2018

日本電気株式会社の許可なく複製・改変などを行うことはできません。

本書の内容に関しては将来予告なしに変更することがあります。