FUJITSU

PCIe SSD-365GB

PCIe SSD-785GB

PCIe SSD-1.2TB

Microsoft Windows 向け
ioMemory VSL ユーザー ガイド

2013 年 9 月

富士通株式会社

著作権および商標

Copyright © 2013 Fujitsu Technology Solutions GmbH.

All rights reserved.

お届けまでの日数は在庫状況によって異なります。技術的修正の権利を有します。

使用されているハードウェア名およびソフトウェア名は、各社の商標です。

- 本書の内容は、改善のため事前連絡なしに変更することがあります。
- 本書に記載されたデータの使用に起因する、第三者の特許権およびその他の権利の侵害については、当社はその責を負いません。
- 無断転載を禁じます。

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

Intel、インテルおよび Xeon は、米国およびその他の国における Intel Corporation またはその子会社の商標または登録商標です。

Fusion-io、Fusion-io のロゴ、ioMemory、Virtual Storage Layer、VSL、Flashback、Flashback Protection、ioManager、および ioDrive は、Fusion-io, Inc.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。

このドキュメントに記載されているその他の組織および製品の名称は、各所有者の商標またはサービス マーク（該当する場合）です。明記されていない限り、このドキュメントに記載されている他のいかなる組織や製品とも一切関係ありません。

本書をお読みになる前に

安全にお使いいただくために

本書には、本製品を安全に正しくお使いいただくための重要な情報が記載されています。

本製品をお使いになる前に、本書を熟読してください。特に、添付の『安全上のご注意』をよくお読みになり、理解されたうえで本製品をお使いください。また、『安全上のご注意』および当マニュアルは、本製品の使用中にいつでもご覧になれるよう大切に保管してください。

電波障害対策について

この装置は、クラスA 情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。

VCCI-A

アルミ電解コンデンサについて

本製品のプリント板ユニットやマウス、キーボードに使用しているアルミ電解コンデンサは寿命部品であり、寿命が尽きた状態で使用し続けると、電解液の漏れや枯渇が生じ、異臭の発生や発煙の原因になる場合があります。

目安として、通常のオフィス環境（25 ℃）で使用された場合には、保守サポート期間内（5年）には寿命に至らないものと想定していますが、高温環境下での稼働等、お客様のご使用環境によっては、より短期間で寿命に至る場合があります。寿命を超えた部品について、交換が可能な場合は、有償にて対応させていただきます。なお、上記はあくまで目安であり、保守サポート期間内に故障しないことをお約束するものではありません。

ハイセイフティ用途での使用について

本製品は、一般事務用、パーソナル用、家庭用、通常の産業用等の一般的用途を想定して設計・製造されているものであり、原子力施設における核反応制御、航空機自動飛行

制御、航空交通管制、大量輸送システムにおける運行制御、生命維持のための医療器具、兵器システムにおけるミサイル発射制御など、極めて高度な安全性が要求され、仮に当該安全性が確保されない場合、直接生命・身体に対する重大な危険性を伴う用途（以下「ハイセイフティ用途」という）に使用されるよう設計・製造されたものではございません。お客様は、当該ハイセイフティ用途に要する安全性を確保する措置を施すことなく、本製品を使用しないでください。ハイセイフティ用途に使用される場合は、弊社の担当営業までご相談ください。

瞬時電圧低下対策について

本製品は、落雷などによる電源の瞬時電圧低下に対し不都合が生じることがあります。電源の瞬時電圧低下対策としては、交流無停電電源装置などを使用されることをお勧めします。

（社団法人電子情報技術産業協会(JEITA) のパーソナルコンピュータの瞬時電圧低下対策ガイドラインに基づく表示）

外国為替及び外国貿易法に基づく特定技術について

当社のドキュメントには「外国為替及び外国貿易法」に基づく特定技術が含まれていることがあります。特定技術が含まれている場合は、当該ドキュメントを輸出または非居住者に提供するとき、同法に基づく許可が必要となります。

目次

はじめに 9
概要 9
ioMemory デバイスについて 9
ioMemory Virtual Storage Layer（VSL）について 9
Flashback Protection テクノロジーについて 10
ソフトウェアのインストール 11
インストール概要 11
新規 ioMemory VSL インストール 11
既存の ioMemory VSL インストール 13
サイレント インストール オプション 14
コンポーネントのインストール 15
サイレント アンインストール 15
古いファームウェアのチェック 15
コマンドライン インターフェイスを使用したチェック 15
オプションの GUI インターフェイスを使用したチェック 15
PCIe 電力の有効化 16
デバイスの名前付け 16
ファイル システムの追加 17
RAID 構成の作成 18
デバイスのスワップとしての使用 18
TRIM のサポートについて 19
Windows Server 2008 R2 における TRIM 19
Windows Server 2003 および Windows Server 2008 R1 における TRIM 20
メンテナンス 21
デバイスの LED インジケーター 21
SNMP のサポート 22
GUI 管理 23
コマンドライン ユーティリティ 23
PCIe Power Override の有効化 23

電源オーバーライド パラメータの有効化 25
ioMemory VSL のアンインストール 26
ioMemory VSL のアップグレード（RAID 構成なし） 27
ioMemory VSL のアップグレード（RAID 構成あり） 28
デバイスのファームウェアのアップグレード 28
ファームウェア バージョンの表示 29
アップグレードの実行 29
VMware のゲスト OS のアップグレード 29
最適化 30
突然のシャットダウンの問題 30
自動接続（auto-attach）の無効化 30
自動接続（auto-attach）の有効化 31
性能とチューニング 32
DVFS の無効化 32
ACPI C ステートの制限 32
NUMA アフィニティの設定 33
割り込みハンドラーのアフィニティの設定 33
付録 A – イベント ログ メッセージのトラブルシューティング 34
エラー メッセージ 34
情報メッセージ 35
付録 B – 手動インストール 36
Windows Server 2003 での手動インストール 36
インストール ウィザード 38
Windows Server 2008 での手動インストール 45
インストール ウィザード 47
付録 C – コマンドライン ユーティリティ 48
fio-attach 49
fio-beacon 50
fio-bugreport 50
fio-config 51
fio-detach 55

fio-format 56
fio-pci-check 57
fio-status 58
fio-sure-erase 61
クリアのサポート 62
パージのサポート 63
fio-trim-config 63
fio-update-iodrive 64
付録 D – TRIM のサポート 66
はじめに 66
プラットフォーム 66
TRIM サービスの使用 67
TRIM の開始と停止 67
TRIM の有効化 67
TRIM の即応性（アグレッシブネス）の制御 68
構成 68
付録 E – ioMemory デバイスの正常性の監視 70
NAND フラッシュおよびコンポーネントの障害 70
正常性ステータスの指標 70
正常性（ヘルスモニターリング）の監視手法 71
ソフトウェア RAID と正常性監視 72
付録 F – ioMemory デバイスでの Windows ページ ファイルの使用 73
はじめに 73
デバイスでのページングのサポートの構成 73
ioMemory VSL による RAM の消費 73
非ページ メモリー プール 74
ページングのサポートの有効化/無効化 75
Windows のページ ファイルの管理 76
ページング ファイルのセットアップ 76
システム ドライブでのページング ファイルの構成 78
最小コミット可能メモリーの確保 80

ページ ファイルの動作の確認 80
性能 80
付録 G – SNMP のテスト モードおよび MIB のサポート 81
はじめに 81
テスト モードのレジストリ値の使用 82
SNMP MIB のサポート 84
付録 H – SMI-S インターフェイス 86
SMI-S インターフェイスの概要 86
Windows への SMI-S WMI プロバイダーのインストール 87
予期される警告メッセージ 87
Windows における SMI-S インストールの確認 88
手動登録 92
解説 93
実装 95
通知 102
付録 I – NUMA 構成 106
NUMA アーキテクチャについて 106
FIO_AFFINITY パラメータ 106
高度な構成の例 107
エラーがないかログを確認する 109
付録 J – VSL 2.x から 3.x へのデバイスのアップグレード 110
アップグレード手順 111
付録 K – 付属資料の使用許諾 114
AVR Bootloader 114
tree.h 114

はじめに

概要

富士通のソリッドステート ストレージ デバイスをご購入いただきありがとうございます。このガイドでは、ioMemory デバイス用ソフトウェアのインストール、トラブルシューティング、およびメンテナンスの方法について説明します。

このマニュアルで **ioMemory デバイス**と記載されている個所は、実際にお使いのデバイス（ioDrive2 デバイス、または ioDrive Duo デバイスを構成する 2 つの ioMemory デバイスのいずれかなど）に置き換えてお読みください。

ioMemory デバイスについて

ioMemory デバイスは、革新的なシリコンベースのストレージ アーキテクチャに基づいて設計された世界最先端の NAND フラッシュ ストレージ デバイスであり、DRAM に匹敵する性能と、今日のハード ディスクと同等のストレージ容量を備えているため、ストレージの性能が飛躍的に高まります。ioMemory デバイスにより、あらゆるコンピューターにおいてエンタープライズ SAN をしのぐ I/O 性能を発揮できます。

ioMemory デバイスは、I/O 負荷の高いアプリケーションにおける帯域幅性能の改善を目的として設計されたデータ アクセラレーターです。ioMemory デバイスは、今日のデータ センターが直面している難しい課題に対して理想的な解決策を提示する、他に類のないソリューションです。

ioMemory Virtual Storage Layer（VSL）について

ioMemory® Virtual Storage Layer™（VSL: 仮想ストレージ レイヤー）は、単なるハードウェア ドライバーではなく、ioMemory デバイスが優れた性能を発揮するために必要不可欠な技術です。VSL™は、RAM 仮想化サブシステムとディスク I/O サブシステムの両方の利点を取り入れた、ハイブリッド型のサブシステムです。一見すると、ブロックベースのアプリケーションやソフトウェアとのインターフェイスとして動作するディスクのように見えますが、実際には、最大の性能を発揮するために RAM のような動作をしています。これにより、次のような大きなメリットがあります。

- 性能: VSL では、複数の CPU コアに直接かつ並列にアクセスできるため、ほぼリニアな性能スケーリングが可能で、さまざまな読み書き負荷において安定した性能を発揮できます。また、割り込みやコンテキスト スイッチも最小限に抑えられるため、レイテンシーが短くなります。
- 拡張性: VSL を利用すると、フラッシュ最適化に対応したソフトウェア開発が可能になります。個々の ioMemory モジュールを、フラッシュ最適化されたデータ センター構築のための構成要素として柔軟に使用できます。

単なる追加ディスク ドライブ以上の役割を果たすフラッシュ ストレージ

他社提供の SSD では、PCIe カード型でも、ドライブ ベイ型でも、フラッシュを RAID コントローラー管理下の単なる追加のディスク ドライブとして扱っています。このアプローチには、次のような制約があります。

- 性能と信頼性が低下する
- レイテンシーが長くなり、複雑性が増す
- フラッシュ ストレージ メディアに関連するソフトウェア開発および最適化で発揮できる性能が制限される

ioMemory デバイスは、ioMemory VSL を利用することによって、このような制約のあるアプローチを回避し、性能向上と最適化の可能性を広げています。

Flashback Protection テクノロジーについて

NAND フラッシュは、他の多くのメモリー デバイスと同様、長年使用するうちにいつかは障害が発生します。発生する障害には、永続的なものと一時的なものがあります。Fusion Flashback™の冗長性機能は、永続的な障害が発生している ioMemory チップへの対策として設計されたものであり、ソフト エラーに対して ECC（誤り訂正符号）を超える保護機能を提供します。

Flashback テクノロジーは、RAID によく似たリアルタイムの冗長性をチップレベルで提供し、容量や性能を犠牲にすることなく耐障害性を実現します。物理的な RAID スキームを使用して冗長性や保護を実現しようとするソリューションでは、一般に、容量（RAID 1）または性能（RAID 5）を犠牲にせざるを得ません。

自己修復特性を備えた Fusion Flashback Protection™テクノロジーでは、他のどのフラッシュ ソリューションよりも高い性能、低い障害発生率、長い耐用性が保証されます。

ソフトウェアのインストール

このソフトウェアのインストールに進む前に、次の内容を確認してください。

1. このリリースの『ioMemory VSL Release Notes』(英文)の「supported operating systems」の一覧に、使用するオペレーティング システムが含まれていることを確認します。
2. ioMemory VSL をインストールする前に、ioMemory デバイスが適切に取り付けられていることを確認します。詳細およびハードウェア要件については、『ioMemory ハードウェア インストール ガイド』を参照してください。

インストール概要

1. VSL 2.x 用に構成された ioDrive デバイスが存在するシステムにこのバージョンの ioMemory VSL をインストールする場合は、「付録 J – VSL 2.x から 3.x へのデバイスのアップグレード」の手順に慎重に従う必要があります。

   VSL 3.x.x 用のファームウェアにデバイスをアップグレードする必要はないが、システムに以前のバージョンの ioMemory VSL がインストールされている場合は、ioMemory VSL ソフトウェアをアンインストールする必要があります。手順については、「既存の ioMemory VSL インストール」を参照してください。ソフトウェアをアンインストールした後に、引き続きこの参照先ページの手順に従います。

2. 最新バージョンの ioMemory VSL をインストールします。

   トラブルシューティングの目的でインストール ログをキャプチャする方法の詳細については、マイクロソフトのサポート技術情報の記事を参照してください。

3. 必要に応じて、ファームウェアを最新バージョンにアップグレードします(アップグレードすることをお勧めします。「古いファームウェアのチェック」を参照)。最新バージョンよりも古いバージョンのファームウェアを使用できる ioDrive2 デバイスがこの手順に該当します。
4. 「ファイル システムの追加」、「RAID 構成の作成」などを行って、デバイスを構成します。

新規 ioMemory VSL インストール

新しいシステムに ioMemory VSL ソフトウェアをインストールするには:

1. 『ioMemory ハードウェア インストール ガイド』で説明されているすべての取り付け手順を完了します。

2. 管理者権限を持つアカウントを使用してコンピューターにログインします。
3. **PRIMERGY ダウンロードページ:** **http://jp.fujitsu.com/platform/server/primergy/downloads/**から Windows 用の ioMemory VSL インストール プログラムをダウンロードして、デスクトップまたはその他の任意のディレクトリに保存します。
4. ioMemory VSL インストール プログラムを実行します。インストール プログラムでは、インストールのオプションを示すカスタム セットアップ ツリービューが表示されます。
5. ドロップダウン メニューからコンポーネントを選択して、インストールのタイプを選択します。後でインストールのタイプを変更する場合は、コントロール パネルで、[プログラムと機能]の[修復]オプション、または[プログラムの追加と削除]を使用できます。

各コンポーネントをクリックすると、説明が表示されます。説明は、インストール ツリーの右側に表示されます。

6. [Next]をクリックします。
7. インストール用に異なるフォルダーを選択するには、フォルダーを参照して、[OK]をクリックします。デフォルトのフォルダーは `C:\Program Files\Fusion-io ioMemory VSL` です。
8. 画面上のプロンプトに従って、インストールを完了します。
9. インストーラーの完了画面で、[Finish]をクリックします。

インストール プロセスを完了するためにシステムを再起動するように求めるプロンプトが表示されます。再起動を求めるプロンプトが表示されない場合でも、インストールが完了したらシステムを再起動してください。再起動後に Windows が ioMemory デバイスを認識しない場合は、デバイスの ioMemory VSL を手動でインストールする必要があります。手動インストールの詳細については、「付録 B – 手動インストール」を参照してください。

インストール プログラムによって、次の処理が行われます。

- ソフトウェア コンポーネント用のフォルダーが作成されます(デフォルトのパスは `C:\Program Files\Fusion-io ioMemory VSL` です)。
- ioMemory VSL のインストールおよび読み込みが行われます(再起動が必要となる場合があります)。
- SNMP のサポートがインストールされます(SNMP のインストールが選択され、マイクロソフトの SNMP がインストールされており、SNMP サービスが実行されている場合)。
- VSL ユーティリティ用のフォルダーが作成されます。デフォルトのパスは `C:\Program Files\Common Files\VSL Utils` です。

インストール プログラムによってドライブ上に ioMemory VSL のフォルダーが作成されるときに、次のサブフォルダーも作成されます。

- `<VSL バージョン>\Driver` – デバイス マネージャーを使用した手動インストール用のフォルダーです。

- Firmware – ioMemory デバイスの最新のファームウェアが含まれています。
- SNMP – SNMP コンポーネントが含まれています。
- SMIS および SDK – インストールが選択された場合に作成されます。

ioManager は、ioMemory デバイスを管理するための無料のブラウザーベース ソフトウェアです。**PRIMERGY ダウンロードページ:http://jp.fujitsu.com/platform/server/primergy/downloads/**で入手することもできますが、ioMemory VSL パッケージおよびドキュメントとは異なる場所にあります。

続いて、「古いファームウェアのチェック」に進みます。

既存の ioMemory VSL インストール

以前に ioDrive デバイスが取り付けられているシステムでは、最初に「付録 J – VSL 2.x から 3.x へのデバイスのアップグレード」の手順を完了してから新しい ioDrive2 デバイスを取り付けてください。

既存のインストールに最新の ioMemory VSL Windows ソフトウェアをインストールするには、次の手順に従います。

1. このバージョンのソフトウェアに用意されているリリース ノート ファイルで、インストールを完了するために必要となる可能性がある追加の手順について確認します。
2. 管理者として、または管理者権限を持つアカウントを使用してログインします。
3. (Windows のバージョンに応じて)コントロール パネルの[プログラムと機能]または[プログラムの追加と削除]を使用して、既存の VSL やユーティリティなどをアンインストールします。
4. コンピューターを再起動します。

   ioMemory VSL インストール プログラムでは、以前のバージョンのソフトウェアの削除が試みられます。ただし、インストール プログラムで削除に失敗し、最新バージョンのインストール後にユーザーが以前のバージョンを削除した場合は、再起動しても ioMemory VSL は読み込まれません。この場合は、a)コントロール パネルの[プログラムと機能](または[プログラムの追加と削除])からインストール プログラムの修復オプションを実行して、b)コンピューターを再起動する必要があります。

5. **PRIMERGY ダウンロードページ:http://jp.fujitsu.com/platform/server/primergy/downloads/**でからデスクトップまたは任意のディレクトリに Windows 用の VSL インストール プログラムをダウンロードします。
6. ioMemory VSL インストール プログラムを実行します。インストール プログラムでは、インストールのオプションを示すカスタム セットアップ ツリービューが表示されます。

7. ドロップダウン メニューからコンポーネントを選択して、インストールのタイプを選択します。後でインストールのタイプを変更する場合は、コントロール パネルで、[プログラムと機能]の[修復]オプション、または[プログラムの追加と削除]を使用できます。

   各コンポーネントをクリックすると、説明が表示されます。説明は、インストール ツリーの右側に表示されます。

8. [Next]をクリックします。
9. インストール用に異なるフォルダーを選択するには、フォルダーを参照して、[OK]をクリックします。デフォルトのフォルダーは `C:\Program Files\Fusion-io ioMemory VSL` です。
   - インストーラーによって、VSL ユーティリティ用のフォルダーも作成されます。デフォルトのパスは `C:\Program Files\Common Files\VSL Utils` です。
10. 画面上のプロンプトに従って、インストールを完了します。
11. インストーラーの完了画面で、[Finish]をクリックします。

インストール プロセスを完了するためにシステムを再起動するように求めるプロンプトが表示されます。再起動を求めるプロンプトが表示されない場合でも、インストールが完了したらシステムを再起動してください。再起動後に Windows が ioMemory デバイスを認識しない場合は、デバイスの ioMemory VSL を手動でインストールする必要があります。手動インストールの詳細については、「付録 B – 手動インストール」を参照してください。

システムが再起動したら、続いて「古いファームウェアのチェック」に進みます。

サイレント インストール オプション

以前のバージョンのアンインストール
以前のバージョンの ioMemory VSL がインストールされている場合は、最初に以前のバージョンをアンインストールする必要があります（後の「サイレント アンインストール」を参照）。サイレント インストール オプションを使用して新しいバージョンをインストールした後、手動でコンピューターを再起動する必要があります。ioMemory VSL のユーティリティまたは機能を使用する前に再起動する必要があります。

リモートからインストールする場合や、スクリプトを使用してインストールする場合は、コマンドライン インターフェイスでインストール プログラムを実行するときにサイレント インストール オプション（`/quiet`）を使用できます。

コマンドライン インターフェイスで、`.exe` インストーラー ファイルがあるフォルダーに移動して、次のコマンドを実行します。

```
<インストール名>.exe /quiet
```

<インストール名>.exe は、インストーラー ファイルの名前です。

このオプションを使用すると、ioMemory VSL がデフォルト設定を使用してインストールされるため、インストール中に[Next]をクリックしたり、設定を選択したりする必要がありません。

コンポーネントのインストール

デフォルトのコマンドライン インストールでは、ioMemory VSL（コマンドライン ユーティリティを含む）およびファームウェア ファイルがインストールされます。次のコマンドを使用して、コンポーネントを追加できます。

```
<インストール名>.exe /quiet ADDLOCAL=SNMP,SDK,SMIS
```

インストールしないコンポーネント（および前後のコンマ）は、この記述から削除します。

サイレント アンインストール

次のコマンドを使用して、ioMemory VSL をサイレント アンインストールできます。

```
<インストール名>.exe /uninstall /quiet
```

古いファームウェアのチェック

コマンドライン インターフェイスを使用したチェック

ここで示すコマンドライン ユーティリティの詳細については、「付録 C – コマンドライン ユーティリティ」を参照してください。

1. `fio-status` を実行して、出力を確認します。
   - デバイスが最小モード（minimal mode）の場合は、ファームウェアが古いことを示しています。
   - デバイスに対して表示されるファームウェアの番号が、リリース ノートに記載されている最新のファームウェア バージョンよりも小さい場合、ファームウェアは最新ではありませんが、古くはありません。
2. ファームウェアが最新ではない場合や古い場合は、`fio-update-iodrive` ユーティリティを使用して更新します。

オプションの GUI インターフェイスを使用したチェック

（オプションの）ioManager GUI プログラムを使用して、古いファームウェアをチェックできます。

古いファームウェアまたは最新ではないファームウェアをチェックするには:

1. ioManager を起動して、警告の記号が表示されているデバイスを探します。
2. 警告の記号が表示されている任意のデバイスをクリックして、ファームウェアが古いためにこのアラートが表示されていることを確認します。
3. ファームウェアの更新が必要なすべてのデバイスを選択し、ioManager を使用してファームウェアを更新します。詳細については、『ioManager ユーザー ガイド』を参照してください。

最新ではないが古くはないファームウェアをチェックするには:

1. リリース ノートに記載されている最新のファームウェア バージョン名を確認します。
2. ioManager を使用して、各 ioMemory デバイスのファームウェア バージョンを最新と比較します。
3. ファームウェアの更新方法については、『ioManager ユーザー ガイド』を参照してください。

PCIe 電力の有効化

ioDrive2 Duo デバイスなどのデュアル ioMemory デバイスを取り付けている場合、それらのデバイスが適切に機能するには、追加の電力（PCIe Gen2 スロットによって供給される最低限の電力である 25 W を超える電力）を必要とする場合があります。

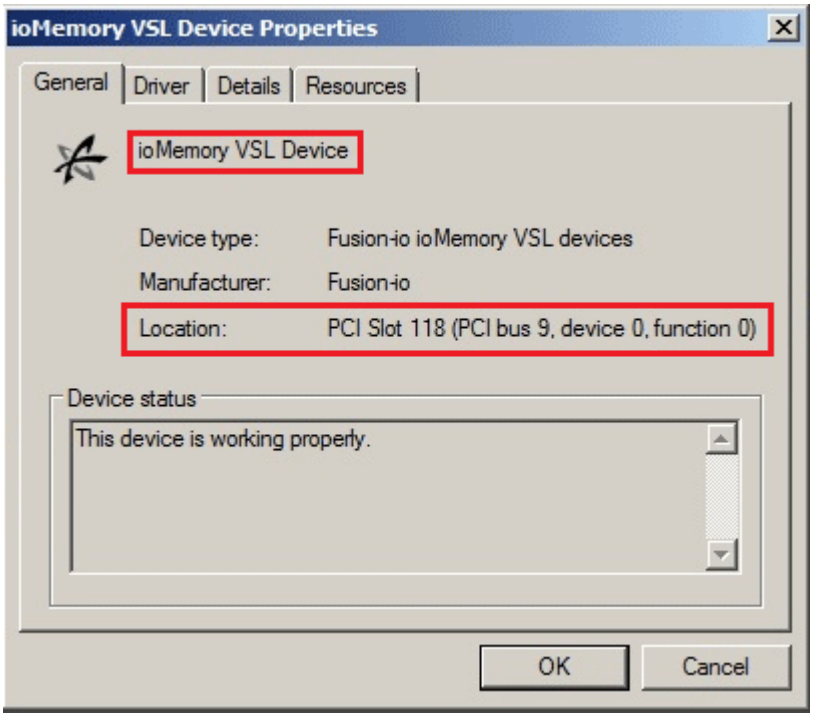
追加の電力は、電源ケーブル（『ioMemory ハードウェア インストール ガイド』を参照）または PCIe スロットから供給できます。PCIe スロットからデバイスに追加の電力を供給できるようにする方法については、「メンテナンス」の「PCIe Power Override の有効化」を参照してください。

デバイスの名前付け

各デバイスを特定するために、インストール プロセスの一環として ioMemory デバイスに名前と番号が割り当てられます。構文は、`fctx` です。x は、ioDrive を取り付けた PCIe バスの番号です。このバス番号を表示するには、ioManager を使用するか、または次の手順に従います。

1. [スタート]ボタン、[コントロール パネル]、[システム]、[ハードウェア]、[デバイス マネージャ]の順にクリックします。
2. [Fusion ioMemory VSL devices]をクリックします。

3. 一覧内の ioMemory デバイスをクリックします。プロパティ ダイアログ ボックスが表示されます。



[Location]フィールドに、デバイスの PCIe バス番号（この場合は `fct9`）が表示されます。

バス番号には、システムの製造業者が 0 以上の番号を割り当てます。これらの番号は、必ずしもバスの物理的な場所を反映していません（たとえば、マザーボードの端から 2 番目のスロットはバス 2 の場合もありますが、バス 16 やその他の任意の番号の場合もあります。取り付けたデバイスの具体的なバス番号は、デバイス マネージャーで確認できます。ioManager を使用してこの番号を表示することもできます）。

ファイル システムの追加

ioMemory デバイスを取り付け、ioMemory VSL をインストールしたら、Windows のディスクの管理ユーティリティを使用して、アプリケーションからデバイスを利用できるようにします。通常、新しいデバイスは、Windows によって検出および初期化されて、ディスクの管理に表示されます。その後、Windows 標準の手順を使用して、ioMemory デバイスでパーティションの追加、ボリュームのフォーマット、または RAID 構成の作成を実行できます（詳細については、Windows のディスクの管理ユーティリティのドキュメントを参照してください）。

デバイスが Windows によって初期化されない場合は、手動で初期化できます。ioMemory デバイスを初期化するには、次の手順に従います。

1. [スタート]ボタン、[コントロール パネル]の順にクリックします。
2. [管理ツール]をクリックします。
3. [コンピュータの管理]をクリックします。
4. コンソール ツリーの[記憶域]セクションの[ディスクの管理]をクリックします。
5. 右側に表示されたストレージ デバイスの一覧で ioMemory デバイスを検索し、右クリックします（一覧に ioMemory デバイスが表示されない場合は、[操作]メニューの[ディスクの再スキャン]をクリックします。一覧に ioMemory デバイスを表示するためにコンピューターの再起動が必要になる場合もあります）。
6. [ディスクの初期化]をクリックします。

これで、ディスクの管理ユーティリティを使用して、ioMemory デバイスにファイル システムを追加できるようになりました。

RAID 構成の作成

1 つ以上の ioMemory デバイスを追加して、ioMemory デバイスを RAID 構成の一部として使用できます。このためには、ioMemory デバイスをダイナミック ボリュームとしてフォーマットする必要があります。その後、これらのダイナミック ボリュームを使用して、複数ディスクの RAID 構成を作成できます（スパン、ストライプ、ミラー、または RAID 5）。

RAID 構成を行う具体的な手順の詳細については、Windows のディスクの管理ユーティリティのドキュメントを参照してください。

RAID 1/ミラーリングを使用しており、一つのデバイスで障害が発生した場合は、RAID を再構築する前に、（もう一つの障害が発生していないデバイスではなく）交換したデバイスで fio-format を実行してください。

デバイスのスワップとしての使用

ioMemory デバイスをスワップ領域として安全に使用するには、fio-config ユーティリティを使用して、特別な事前割り当てパラメータを渡す必要があります。

例:

```
fio-config -p FIO_PREALLOCATE_MEMORY 1072,4997,6710,10345
```

- 1072、4997、6710、10345 は、fio-status によって取得したデバイスのシリアル番号です（アダプターのシリアル番号は使用しないでください）。

スワップでは、4K のセクター サイズのフォーマットが必要です。これにより、ソフトウェアによるメモリーの占有領域を適切なレベルに抑えることができます。デバイスのセクター サイズの変更の詳細については、`fio-format` ユーティリティを参照してください。

アダプターではなく、ioMemory デバイスのシリアル番号を指定してください。

デバイスをスワップ領域として使用可能にするためには、`FIO_PREALLOCATE_MEMORY` が必要です。これにより、デバイスの動作中にクラッシュが発生しなくなります。このパラメータの設定の詳細については、付録 I の fio-config のオプションについての説明を参照してください。

ioMemory デバイスでスワップとして使用するための事前割り当てを有効にするには、十分な RAM が利用できる必要があります。十分な RAM がないまま事前割り当てを有効にして ioMemory デバイスを接続すると、ユーザー プロセスが強制終了したり、システムが不安定になったりする場合があります。

このバージョンの ioMemory VSL の RAM 要件については、リリース ノートを参照してください。

`FIO_PREALLOCATE_MEMORY` パラメータは ioMemory VSL によって読み込み時に認識されますが、要求されたメモリーは、実際には指定されたデバイスが接続されるまでは割り当てられません。

TRIM のサポートについて

このバージョンの ioMemory VSL では、多くのオペレーティング システムで TRIM（Discard とも呼ばれます）がデフォルトで有効になっています。

TRIM は、ソリッドステート ストレージに特有の問題に対処するための機能です。ユーザーがファイルを削除した場合、デバイスでは、その領域が再利用可能であることが認識されません。デバイスでは、データが引き続き有効であるとみなされます。

TRIM は、比較的新しいオペレーティング システムに備わっている機能です。TRIM によって、有効なユーザー データが含まれなくなった論理セクターがデバイスに通知されます。ウェアレベリング ソフトウェアは、この機能によって、将来の書き込み操作の処理用にその領域を確保しておき、後で再利用できます。

Windows における TRIM のサポートの詳細については、「付録 D – TRIM のサポート」を参照してください。

Windows では、TRIM と RAID 5 構成を併用することはできません。

Windows Server 2008 R2 における TRIM

Windows Server 2008 R2 には、TRIM のサポートが組み込まれています。ioMemory デバイスは、このオペレーティング システムでは Windows の TRIM コマンドをデフォルトで使用します。

Windows Server 2003 および Windows Server 2008 R1 における TRIM

Windows Server 2003 または Windows Server 2008 R1 には、Windows の TRIM 機能は組み込まれていません。ただし、Windows 用の ioMemory VSL と共に Fusion-io の TRIM サービスがインストールされ、このサービスが必要な TRIM 操作を提供します。

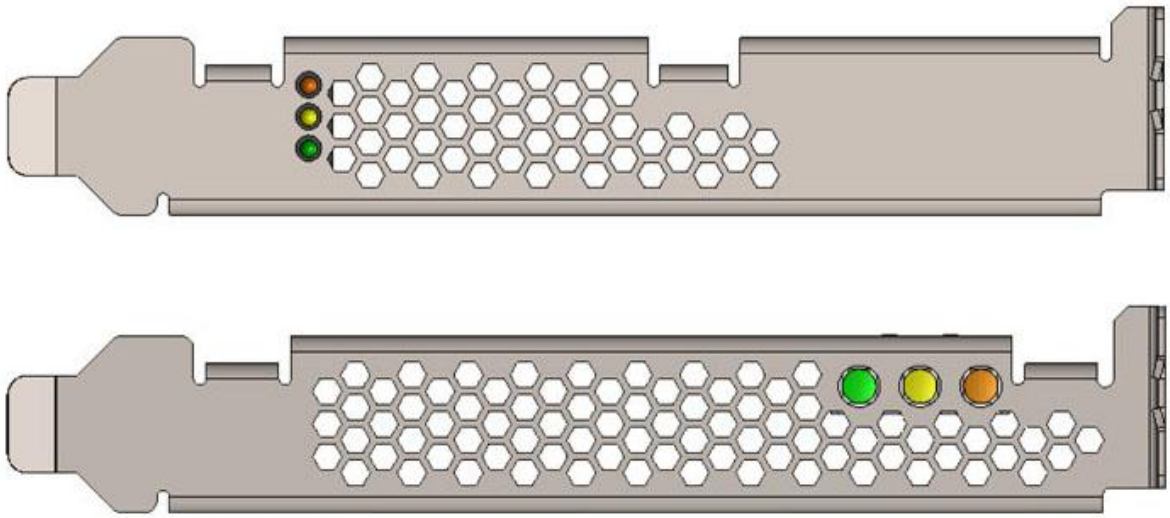
Fusion-io の TRIM サービスは、TRIM をサポートするオペレーティング システム（Windows Server 2008 R2 など）が検出されない場合にデフォルトで有効になります。`fio-trim-config` ユーティリティを使用して、Fusion-io の TRIM サービスを無効にできます。詳細については、「fio-trim-config」を参照してください。

メンテナンス

ioMemory VSL には、デバイスのメンテナンス用のソフトウェア ユーティリティが含まれています。監視用のオプションとして、SNMP をインストールすることもできます。

デバイスの LED インジケーター

ioMemory デバイスには、ドライブのアクティビティやエラー状態を示す 3 つの LED が用意されています。デバイスの LED は、次に示す図のいずれかに似た構成になっています。



これらの LED が示す情報について次の表で説明します。

Green	Yellow	Amber	Indicates	Notes
○	○	○	Power off	
○	○	●	Power on (Driver not loaded and device not attached)	Load Driver and attach device
●	○	○	Power on, Driver loaded (device may not be attached)	You may need to attach the device
●	(Flashing) ◔	○	Writing (Rate indicates volume of writes)	Can appear in combination with the Read LED indication
(Flashing) ◔	○	○	Reading (Rate indicates volume of reads)	Can appear in combination with the Write LED indication
●	●	●	Location Beacon	

ioFX デバイスには、ioFX のロゴがイルミネーション表示される追加の LED があります。この LED には機能的な意味はなく、消灯することもできません。

SNMP のサポート

ioMemory VSL の Windows 用セットアップ プログラムには、SNMP のサポートをインストールするオプションが用意されています。このオプションを選択した場合、セットアップ プログラムによってコンポーネントがインストールされて、SNMP のサポート用にレジストリが変更されます。また、報告を受信するコンピューターに Microsoft Windows の SNMP サービスがインストールされ、サービスが実行されている必要があります。

VSL の Windows 用セットアップ プログラムを実行すると、VSL のエージェントを認識するために Windows の SNMP サービスが停止され、再度開始されます。

セットアップ時に SNMP のサポートのインストールを選択しておらず、後でインストールする場合は、セットアップ プログラムを再実行します。この場合、項目の一覧から SNMP のサポートのみのインストールを選択します。セットアップ プログラムによるインストールが完了した後、Windows の SNMP サービスが停止され、再度開始されます。

SNMP テスト モードの使用の詳細については、「付録 G – SNMP のテスト モードおよび MIB のサポート」を参照してください。

GUI 管理

ioManager は、ioMemory デバイスを管理するための無料のブラウザーベース ソリューションです。http://support.fusionio.com から入手することもできますが、ドキュメントと共に異なるダウンロードの場所にあります。

ioManager では、次の処理を実行できます。

- ファームウェアのアップグレード
- ローレベル フォーマット
- 接続および切断操作
- デバイスのステータス情報および性能情報の確認

コマンドライン ユーティリティ

Windows 用セットアップ パッケージには、デバイスを管理するためのいくつかのコマンドライン ユーティリティも含まれています。

- `fio-attach`
- `fio-beacon`
- `fio-bugreport`
- `fio-config`
- `fio-detach`
- `fio-format`
- `fio-pci-check`
- `fio-status`
- `fio-sure-erase`
- `fio-trim-config`
- `fio-update-iodrive`

これらの各ユーティリティの詳細については、「付録 C – コマンドライン ユーティリティ」を参照してください。

PCIe Power Override の有効化

ioDrive Duo デバイスなどの複数の ioMemory デバイスで構成された製品を取り付けている場合、それらのデバイスが適切に機能するには、追加の電力（PCIe Gen2 スロットによって供給される最低限の電力である 25 W を超える電力）を必要とする場合があります。使用するデバイスで追加の電力が必要ではない場合でも、追加の電力を供給するとすべての ioDrive2 Duo デバイスで性能が向上する可能性があります。

ioDrive2 Duo デバイスが適切に機能するためには、追加の電力を供給する**必要があります**。追加の電力を必要とするデバイスの詳細については、『ioMemory ハードウェア インストール ガイド』の「複数のデバイスで構成された製品の電源ケーブル」を参照してください。

追加の電力は、2 つの方法で供給できます。

- **外部電源ケーブル**: このケーブルは、すべての ioDrive2 Duo デバイスに同梱されています。このケーブルの取り付け方法の詳細については、『ioMemory ハードウェア インストール ガイド』を参照してください。

  電源ケーブルを使用した場合は、すべての電力がケーブルから供給されます。PCIe スロットの電力は使用されません。

- **スロットからすべての電力を供給する**: 一部の PCIe スロットは、追加の電力を供給します（多くの場合、最大で 75 W）。スロットの定格電力が 55 W 以上の場合は、VSL モジュールのパラメータを設定することによって、デバイスのすべての電力を PCIe スロットから供給できます。この電源オーバーライド パラメータの有効化の詳細については、次のセクションの説明を参照してください。

  このパラメータによって、PCIe スロットからデバイスに 25 W を超える電力を供給するのを禁止する設定が上書きされます。このパラメータは、（デバイスのシリアル番号を使用して）デバイスごとに有効化されます。設定が上書きされると、各デバイスにおいて、最大の性能を発揮するのに必要な最大 55 W までの電力をスロットから供給できます。

  警告
  必要な電力をスロットから供給できない場合に PCIe スロットからすべての電力を供給する設定を有効化すると、デバイスが正しく機能しなくなったり、サーバーのハードウェアが損傷したりする危険性があります。この電源オーバーライド パラメータの不適切な使用に起因する装置に対するすべての損傷については、お客様の責任となります。このような不適切な使用に起因するいかなる損傷についても Fusion-io は一切の責任を負わないことをここに明記します。電源オーバーライド パラメータの使用について質問や懸念事項がある場合は、Fusion-io のカスタマー サポートにご連絡ください。

この電源オーバーライド パラメータを有効化する前に、使用する各 PCIe スロットの定格電力が、すべてのスロット、デバイス、およびサーバー アクセサリに対して十分な電力を供給できるワット数であることを確認します。スロットの電力の制限について判断するには、サーバーのドキュメント、BIOS インターフェイス、セットアップ ユーティリティ、または `fio-pci-check` を利用します。

サーバーの製造業者への確認
各スロットおよびシステム全体の電力の制限と機能については、サーバーの製造業者に問い合わせてください。

次に、重要な考慮事項を示します。

- 複数の ioDrive2 Duo デバイスを取り付けて、各デバイスに対して電源オーバーライド パラメータを有効にする場合は、マザーボードの定格電力が、使用する各スロットに対して 55 W の電力を供給できるワット数であることを確認します。

  たとえば、一部のマザーボードでは、任意の 1 つのスロットでは最大で 75 W まで安全に供給できますが、複数のスロットで 75 W を供給すると電力が制限されます。このような場合に複数のデバイスを取り付けたときも、サーバーのハードウェアが損傷する危険性があります。PCIe スロット全体で使用可能な電力については、製造業者に問い合わせてください。

- 電源オーバーライド パラメータは、適切に有効化すると、システム内で永続的に保持されます。デバイスを取り外し、同じシステム内の別のスロットに取り付けた場合でも、パラメータが有効化されたデバイスではスロットからすべての電力が供給されます。ただし、定格電力が 55 W 未満のスロットにデバイスを取り付けると、サーバーのハードウェアが損傷する危険性があります。
- この電源オーバーライド パラメータは、サーバーごとの ioMemory VSL ソフトウェアの設定であり、デバイスには保存されません。デバイスを新しいサーバーに移動した場合は、外部電源ケーブルを追加するか、または新しいサーバーでデバイスに対してこの電源オーバーライド パラメータを有効にするまでの間、デバイスに対してデフォルトの 25 W の電力制限が適用されます。新しいサーバーにおける PCIe スロット全体で使用可能な電力については、製造業者に問い合わせてください。

電源オーバーライド パラメータの有効化

シリアル番号の特定

このパラメータを有効化する前に、互換性のあるスロットに挿入する各デバイスの**アダプターのシリアル番号**を特定します。アダプターのシリアル番号を特定するには、`fio-status` コマンドライン ユーティリティを使用します。

シリアル番号のラベル
デバイスに貼付されているアダプターのシリアル番号のラベルを調べて、シリアル番号を特定することもできます。ただし、ベスト プラクティスとして、`fio-status` を実行し、それぞれのシリアル番号がアダプターのシリアル番号であることを確認することをお勧めします。アダプターのシリアル番号のラベルは、すべての ioDrive Duo デバイスおよび ioDrive2 Duo デバイスの背面に貼付されています。ioDrive Duo デバイスでは、PCIe コネクターに接続されたプリント基板に貼付されています。

`fio-status` の使用: `fio-status` コマンドライン ユーティリティを実行します。次に、サンプルの出力を示します。

```
fio-status
...
```

```
Adapter: Dual Controller Adapter
        Fusion-io ioDrive2 DUO 2.41TB, Product Number:F01-001-2T41-CS-0001,
FIO SN:1149D0969
        External Power: NOT connected
        PCIe Power limit threshold: 24.75W
        Connected ioMemory modules:
         fct2:  SN:1149D0969-1121
         fct3:  SN:1149D0969-1111
```

この例では、1149D0969 がアダプターのシリアル番号です。

fio-beacon の使用: 複数のデバイスを取り付けている場合は、fio-beacon ユーティリティを使用して、各デバイスの物理的な場所を確認できます。詳細については、付録のユーティリティについてのドキュメントを参照してください。

パラメータの設定

fio-config コマンドライン ユーティリティを使用して、モジュールのパラメータを設定します。次に例を示します。

```
fio-config -p FIO_EXTERNAL_POWER_OVERRIDE <値>
```

このパラメータの<値>は、アダプターのシリアル番号をコンマで区切ったリストです。たとえば、1149D0969,1159E0972,24589 のように指定します。

このパラメータを設定すると、**以前の値が上書きされます**。リストにシリアル番号を追加する場合は、新しいシリアル番号と以前に入力した番号の両方をリストに含める必要があります。リストをクリアするには、値を指定しないでパラメータを設定します。

-p オプションを指定すると、再起動後もパラメータが永続的に保持されます。パラメータを有効にするには、システムを再起動します。

シリアル番号が適切に設定されたかどうかを確認したり、リストにシリアル番号を追加する前にすでに設定済みの番号を確認したりする場合は、-e オプション（現在のパラメータおよび値の列挙）を使用できます。

ioMemory VSL のアンインストール

ioMemory VSL をアンインストールするには、次の手順に従います。

1. [スタート]ボタン、[コントロール パネル]の順にクリックします。
2. [プログラムと機能]をクリックします。

3. Fusion-io ioMemory VSL のエントリをクリックします。
4. [アンインストール]をクリックします。

Windows によって、ioMemory VSL のフォルダー、およびその中にあるすべてのファイルとフォルダーがアンインストールされます。

この手順では、ioMemory VSL のユーティリティはアンインストール**されません**。新しいバージョンの ioMemory VSL にアップグレードする場合は、これらのユーティリティを手動でアンインストールする必要はありません。ただし、ソフトウェアを完全にアンインストールする場合や、以前のバージョンの ioMemory VSL をインストールすることを計画している場合は、`C:\Program Files\Common Files\VSL Utils` フォルダーとその内容を手動で削除する必要があります。

ioMemory VSL のアップグレード(RAID 構成なし)

アップグレードの実行時にデータが失われないように、各新規リリースに付属しているリリース ノート ドキュメント、およびここに示すインストール手順をお読みください。

RAID 構成なしの場合に ioMemory VSL をアップグレードするには:

1. 前の「ioMemory VSL のアンインストール」の手順に従います。
2. **PRIMERGY ダウンロードページ:**
   http://jp.fujitsu.com/platform/server/primergy/downloads/でから最新のドライバーをダウンロードします。
3. Windows パッケージを解凍または実行して、ファイルを任意のディレクトリにコピーします。
4. [スタート]ボタン、[コントロール パネル]の順にクリックします。
5. [管理ツール]をクリックします。
6. [コンピュータの管理]をクリックします。
7. 左側のコンソール ツリーで、[デバイス マネージャ]をクリックします。
8. [Fusion-io Devices]の項目を展開します(1.2.2 以前のドライバーでは、[System Devices]を選択します)。
9. 目的のデバイスを右クリックします。
10. [Update ioMemory VSL Software]をクリックします。更新された ioMemory VSL をインストールする残りの手順の詳細については、必要に応じて「付録 B – 手動インストール」を参照してください。

これで、アップグレードされた ioMemory VSL を使用して Windows が ioMemory デバイスを検出するようになります。

ioMemory VSL のアップグレード（RAID 構成あり）

アップグレードの実行時にデータが失われないように、各新規リリースに付属しているリリース ノート ドキュメント、およびここに示すインストール手順をお読みください。

RAID 構成ありの場合に ioMemory VSL をアップグレードするには:

1. ioMemory デバイスにアクセスしているすべてのアプリケーションを停止します。
2. ioMemory VSL のユーティリティのフォルダーを開きます（このリリースでのデフォルトの場所は `C:\Program Files\Fusion-io ioMemory VSL\Utils` です）。
3. `AutoAttachDisable.reg` ファイルをダブルクリックして、Windows レジストリにキーを追加します。これで、次回コンピューターを再起動したときに、ioMemory デバイスが自動的に接続されなくなります。
4. Windows の[プログラムの追加と削除]で VSL ソフトウェアをアンインストールします。
5. コンピューターを再起動します。
6. **PRIMERGY ダウンロードページ：http://jp.fujitsu.com/platform/server/primergy/downloads/**でから最新の ioMemory VSL ドライバー パッケージをダウンロードします。
7. ioMemory VSL ドライバー パッケージを解凍して、インストールします。インストールが完了したら、[No]をクリックして、手動での再起動を選択します。
8. VSL のユーティリティのフォルダーを開きます（デフォルトの場所は `C:\Program Files\Common Files\VSL Utils` です）。
9. `AutoAttachEnable.reg` ファイルをダブルクリックして、Windows レジストリのキーをリセットします。これで、次回コンピューターを再起動したときに、ioMemory デバイスが自動的に接続されます。
10. デバイスのファームウェアを更新します。次のセクションの「デバイスのファームウェアのアップグレード」の手順に従います。

ファームウェアのアップグレードが完了した後に、コンピューターを再起動します。次回起動時に VSL チェック ユーティリティが実行されます。

これで、アップグレードされたソフトウェアを使用して Windows が RAID 構成のデバイスを検出するようになります。

デバイスのファームウェアのアップグレード

ファームウェアは、システム イベント ログにファームウェアが古いというレポートが表示された場合、またはカスタマー サポート、リリース ノート、正誤表のドキュメントによって指示された場合にのみアップグレードする必要があります。

ファームウェア バージョンの表示

ファームウェア バージョンは、Windows のイベント ログで確認できます。ファームウェア バージョンは、ioManager および `fio-status` コマンドライン ユーティリティによって報告されます。詳細については、『ioManager ユーザー ガイド』または「付録 C – コマンドライン ユーティリティ」の「fio-status」を参照してください。

アップグレード パス
ioMemory デバイスのアップグレード時には、所定のアップグレード パスがあります。ioMemory デバイスをアップグレードする前に、このリリースの ioMemory VSL のリリース ノートを参照してください。

アップグレードの実行

アップグレードを行う前に、念のためデバイスのデータをバックアップする必要があります。

アップグレードを実行するには、ioManager(『ioManager ユーザー ガイド』を参照)または `fio-update-iodrive` コマンドライン ユーティリティを使用します。

ioMemory デバイスには、最低限必要なファームウェアのラベルが貼付されている場合があります(「MIN FW: XXXXXX」など)。このラベルは、デバイスと互換性がある最小のファームウェア バージョンを示しています。

ioMemory デバイスのファームウェアは絶対にダウングレードしないでください。

既存のデバイスが取り付けられている状態で新しい ioMemory デバイスを取り付ける場合は、新しいデバイスを取り付ける前に、現在取り付けられているすべてのデバイスを利用可能な最新のバージョンのファームウェアおよび ioMemory VSL にアップグレードする必要があります。

アップグレードに関する考慮事項については、このリリースの ioMemory VSL のリリース ノートを参照してください。

VMware のゲスト OS のアップグレード

(VM の DirectPath I/O を使用して)VMware のゲスト OS で ioMemory デバイスを使用している場合は、デバイスをアップグレードした後に、ホストの電源をいったんオフにして、再度オンにする必要があります。仮想マシンを再起動するだけでは、変更内容が適用されません。

最適化

ioMemory デバイスは、最適化の必要がありません。ただし、一部のバージョンの Windows では、スケジュールされたタスクとして自動的に最適化が実行されます。必要に応じて、自動最適化をオフにする必要があります。

突然のシャットダウンの問題

停電その他の事情で突然のシャットダウンが発生した場合は、再起動時に ioMemory デバイスで強制的に整合性チェック（Consistency Check）が実行されることがあります。このチェックの完了には、数分間またはそれ以上の時間がかかることがあります。進行状況は、Windows 起動時にパーセンテージで表示されます。

プロンプトに「Fusion-io Consistency Check（Fusion-io 整合性チェック）」のメッセージが表示されてから 15 秒以内に Esc キーを押すと、この整合性チェック（Consistency Check）を取り消すことができます。ただし、チェックを取り消した場合、チェックが完了するまでの間ユーザーは ioMemory デバイスを使用できません（このチェックは、ioManager の接続（Attach）機能を使用して後で実行できます）。

突然のシャットダウンが発生した場合、ioMemory デバイスに書き込まれたデータが失われることはありませんが、重要なデータ構造が適切にデバイスにコミットされていない可能性があります。この整合性チェック（Consistency Check）では、このようなデータ構造が修復されます。

自動接続（auto-attach）の無効化

ioMemory VSL は、デフォルトで、取り付けられたすべての ioMemory デバイスをオペレーティング システムに自動的に接続（auto-attach）します（ioMemory デバイスは、接続しないとアプリケーションやユーザーが使用できません）。トラブルシューティングや診断のために、自動接続（auto-attach）を無効にできます。

自動接続（auto-attach）を無効にするには:

1. ioMemory VSL のユーティリティのフォルダーを開きます（デフォルトの場所は `C:\Program Files\Fusion-io ioMemory VSL\Utils` です）。
2. `autoattachdisable.reg` ファイルをダブルクリックします。
3. ここでプロンプトが表示された場合は、レジストリの変更を行うことをこのプロンプトで確認します。

これにより、次の場所に `AutoAttach` という新しい DWORD パラメータのレジストリ キーが作成されます。

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\fiodrive\Parameters
```

システムを再起動した後、再度自動接続(auto-attach)を有効化(「自動接続(auto-attach)の有効化」を参照)するまでの間、ioMemory デバイスは自動的に接続されなくなります。

ioMemory VSL の問題のトラブルシューティングが完了したら、ioManager を使用して ioMemory デバイスを接続し、Windows で利用できるようにします。

自動接続(auto-attach)の有効化

「自動接続(auto-attach)の無効化」の手順に従って自動接続(auto-attach)を無効にした後、再度有効にするには:

1. ioMemory VSL のユーティリティのフォルダーを開きます(デフォルトの場所は `C:\Program Files\Fusion-io ioMemory VSL\Utils` です)。
2. `autoattachenable.reg` ファイルをダブルクリックします。
3. ここでプロンプトが表示された場合は、レジストリの変更を行うことをこのプロンプトで確認します。

これにより、レジストリの AutoAttach パラメータがリセットされます。これで、次回 Windows システムを再起動したときに、ioMemory デバイスが自動的に接続されます。

性能とチューニング

ioMemory デバイスは、広い帯域幅と高い IOPS（1 秒当たりの入出力操作数）を提供し、短いレイテンシーを実現するための特別な設計になっています。

ioMemory デバイスは IOPS とレイテンシーを向上させますが、デバイスの性能はオペレーティング システムの設定および BIOS の設定によって制限される可能性があります。ioMemory デバイスの性能を最大限に活用するためには、これらの設定のチューニングが必要になることがあります。

通常、Fusion-io デバイスはそのままでも良好な性能を発揮しますが、このセクションでは、チューニングによって最適な性能を引き出せる可能性があるいくつかの領域について説明します。

DVFS の無効化

動的電圧・周波数制御（DVFS）は、CPU の電圧または周波数を調整して CPU による消費電力を削減する電源管理手法です。これらの手法は、電力を節約し、CPU による熱の発生を抑えるのに役立ちますが、CPU が低電力状態と高性能状態との間で遷移することによって性能への悪影響が生じます。

これらの節電手法は、I/O レイテンシーおよび最大 IOPS に悪影響を与えることが知られています。性能の最大化を目的としてチューニングを行う場合、DVSF を制限するか完全に無効化することで効果を期待できますが、一方で消費電力は増加する可能性があります。

DVFS を使用できる場合は、オペレーティング システムの電源管理機能およびシステムの BIOS インターフェイスで設定を変更できます。DVFS の設定は、多くの場合、オペレーティング システムおよび BIOS 内の Advanced Configuration and Power Interface（ACPI）セクションにあります。詳細については、コンピューターのドキュメントを参照してください。

ACPI C ステートの制限

比較的新しいプロセッサーは、使用率が低いときに低電力モードに切り替わる機能を備えています。このようなアイドル状態は ACPI C ステートと呼ばれます。`C0` ステートは通常の最大電力での動作状態を意味し、それより上位の C ステート（`C1`、`C2`、`C3` など）は低電力状態を意味します。

これらの ACPI C ステートは節電に役立ちますが、I/O レイテンシーおよび最大 IOPS に悪影響を与えることが知られています。通常は、C ステートのランクが上がるにつれて、節電のためにより多くのプロセッサー機能を制限するようになり、プロセッサーを `C0` ステートに復帰するまでの時間が長くかかります。

これらの節電手法は、I/O レイテンシーおよび最大 IOPS に悪影響を与えることが知られています。性能の最大化を目的としてチューニングを行う場合、C ステートを制限するか完全に無効することで効果を期待できますが、一方で消費電力は増加する可能性があります。

プロセッサーで ACPI C ステートを利用できる場合は、通常、BIOS インターフェイス（セットアップ ユーティリティと呼ばれることもあります）でこれらを制限または無効化できます。ACPI C ステートの設定は、Advanced Configuration and Power Interface（ACPI）メニューなどに含まれています。詳細については、コンピューターのドキュメントを参照してください。

NUMA アフィニティの設定

NUMA（Non-Uniform Memory Access）アーキテクチャを備えたサーバーでは、ioMemory デバイスの性能を最大限発揮するために特別なインストール手順が必要です。このようなサーバーには、HP DL580、HP DL980、IBM 3850 サーバーなどがあります。

NUMA アーキテクチャを備えたサーバーでは、システムの起動時に BIOS によって NUMA ノード間で PCIe スロットが均等に配分されない場合があります。各 NUMA ノードには、複数の CPU が含まれています。スロットが均等に配分されないと、作業負荷が高まった場合に、半分以上の CPU がアイドル状態であるにもかかわらず、残りの CPU の使用率が 100%になる可能性があります。このような不均等を回避するために、利用可能な NUMA ノードに対して、ioMemory デバイスを手動で均等に割り当てる必要があります。

NUMA アフィニティの設定の詳細については、「付録 I – NUMA 構成」を参照してください。

割り込みハンドラーのアフィニティの設定

NUMA システムに割り込みを割り当てることによって、デバイスのレイテンシーが影響を受ける場合があります。デバイスの割り込みは、アプリケーションが I/O を発行するのと同じ NUMA ソケットに割り当てることをお勧めします。ユーザー アプリケーションのタスクによってこのソケットの CPU の使用率が非常に高くなる場合は、割り込みをリモート ソケットに移動してシステムの負荷を分散することによって性能を向上できる可能性があります。

多くのオペレーティング システムでは動的な割り込みの割り当てが試みられるため、通常は適切に負荷が分散されます。割り込みの割り当ての手動での調整は、特定のハードウェアにおけるアプリケーションのパフォーマンスについての分析を必要とする高度なオプションです。特定のデバイスの割り込みを特定の CPU に固定する方法の詳細については、オペレーティング システムのドキュメントを参照してください。

付録 A – イベント ログ メッセージのトラブルシューティング

Windows システム イベント ログには、ioMemory デバイスに関する情報、警告、およびエラーの Fusion-io ドライブ メッセージが表示されます。

各 ioMemory デバイスには、0 以上の番号が割り振られています。これらの番号は、デバイスを取り付けた PCIe バスの番号を表しています。デバイスに割り振られた番号を表示するには、`fio-status` ユーティリティまたは ioManager を使用します。

Windows イベント ビューアーを開くには、次の手順に従います。

1. [スタート]ボタンをクリックします。
2. [コンピュータ]をクリックし、右クリックして、[管理]をクリックします。
3. [システム ツール]を展開します。
4. [イベント ビューア]を展開します。
5. [Windows ログ]を展開します。
6. [システム]をクリックします。

エラー メッセージ

次に、一般的なイベント ログのエラー メッセージ、および推奨される解決策を示します。

メッセージ	推奨される解決策
Error: ioDrive(x) firmware is too old. The firmware must be updated.	「デバイスのファームウェアのアップグレード」のファームウェア アップグレード手順に従って、ファームウェアを更新します。
Error: ioDrive initialization failed with error code `0xerrorcode` (where `errorcode` is a number that may vary)	1. Windows 用の ioMemory VSL を再インストールします。 2. ioMemory デバイスを取り外して、再度取り付けます。 3. ioMemory デバイスを取り外して、別の PCIe スロットに挿入します。
Error: ioDrive was not attached. Use the `fio-attach` utility to rebuild the drive.	このエラーは、突然のシャットダウンが発生した後に表示されることがあります。`fio-attach` コマンドライン ユーティリティまたは ioManager を使用して、デバイスを再接続できます。ユーティリティによってデバイスで整合性チェック(Consistency Check)が実行されるため、この接続プロセスには最大 10 分ほどかかることがあります。
Warning: ioDrive was not	ユーザーおよびアプリケーションが ioMemory デバイスを利用できるように

loaded because auto-attach is disabled.	するためには、ioMemory デバイスを Windows オペレーティング システムに接続する必要があります（通常は、起動時に自動的に接続されます）。ioMemory VSL は、この自動接続（auto-attach）プロセスの一環として、Windows レジストリに AutoAttach パラメータがあるかどうかを確認します。このレジストリ パラメータを作成して自動接続（auto-attach）を無効にしている場合、接続操作が完了しません。 接続されていないデバイスを接続するには、次の手順に従います。 1. ioManager を実行します。 2. デバイス ツリーから、接続されていない ioMemory デバイスを選択します。 3. [Attach]をクリックします。 4. 接続操作を確認します。 これで、デバイスが Windows オペレーティング システムに接続されます。起動時の自動接続（auto-attach）を再度有効にする方法については、「メンテナンス」の「自動接続（auto-attach）の有効化」を参照してください。

情報メッセージ

次に、一般的なイベント ログの情報メッセージを示します。

メッセージ	追加情報
Affinity not set for ioMemory VSL device fct119 because either WIN_DISABLE_ALL_AFFINITY is set to true or "SetWorkerAffinity119" does not exist in the registry and WIN_DISABLE_DEFAULT_NUMA_AFFINITY is set to true.	WIN_DISABLE_ALL_AFFINITY が 0 に設定されている場合、ドライバーでは、ドライバーに設定されている割り込みおよびワーカー スレッドのアフィニティが有効化されます。 WIN_DISABLE_ALL_AFFINITY が 1 に設定されている場合、ドライバーでは、すべてのアフィニティの設定が無効化されます。この設定は、他のすべてのアフィニティの設定よりも優先されます。 アフィニティの設定の詳細については、「fio-config」を参照してください。

付録 B – 手動インストール

Windows 用セットアップ プログラムによって、Windows オペレーティング システムに ioMemory VSL およびソフトウェアがインストールされます。ただし、次のような場合に、特定の ioMemory デバイスに対する手動でのソフトウェアのインストールが必要になることがあります。

- （アップグレードを含む）ソフトウェアのインストールの後に fio-status を実行しても ioMemory デバイスが表示されない場合
- ioMemory デバイスが以前に取り付けられており、ioMemory VSL ソフトウェアがすでにインストールされているシステムに、新たに ioMemory デバイスを取り付ける場合

次に示す Windows Server 2003 または Windows Server 2008 用の手順に従います。この手順に従うことによって、特定のデバイスに ioMemory VSL がインストールされます。必要に応じて、各デバイスに対してこの手順を繰り返します。

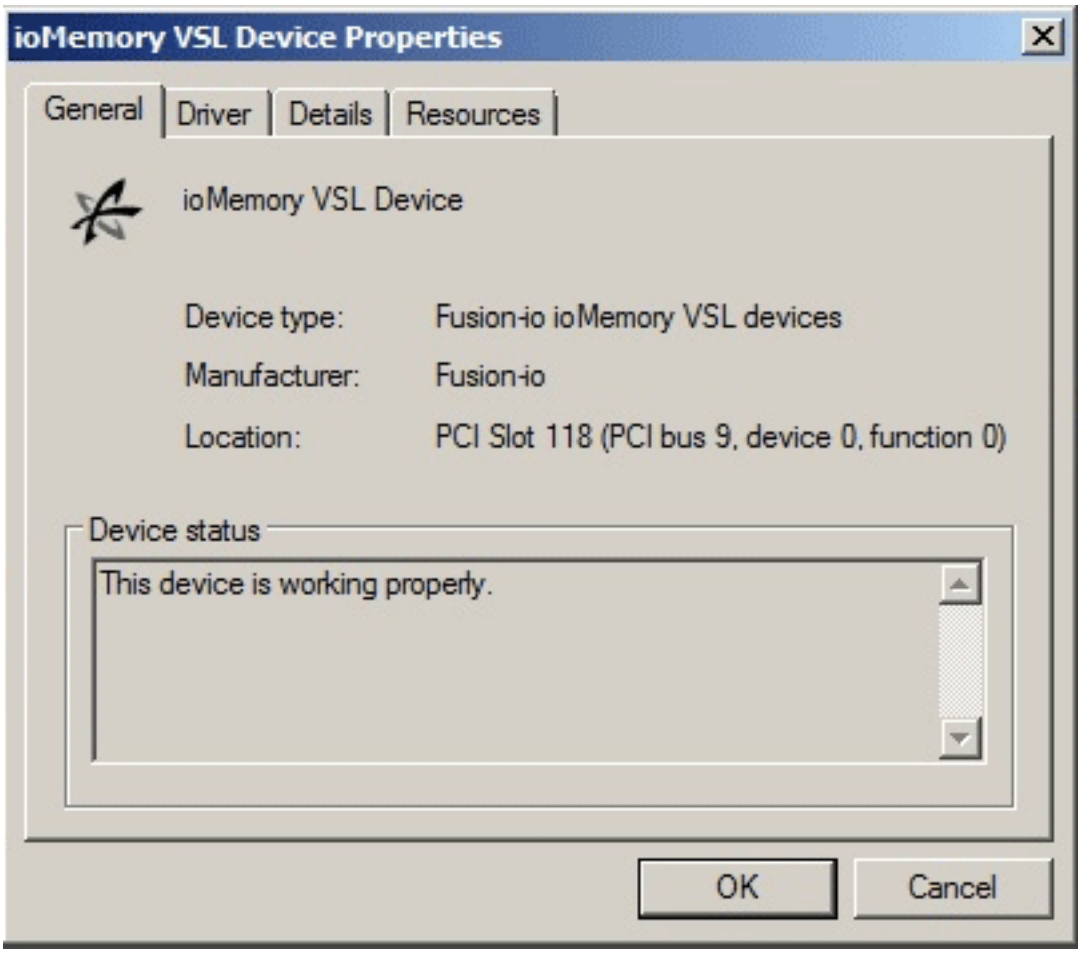
Windows Server 2003 での手動インストール

ioMemory VSL ドライバーを手動でインストールする前に、**PRIMERGY ダウンロードページ:** **http://jp.fujitsu.com/platform/server/primergy/downloads/**でから ioMemory VSL の Windows 用セットアップ プログラムをダウンロードして、実行しておきます。これにより、システムに ioMemory VSL がインストールされ、各 ioMemory デバイスに ioMemory VSL をインストールできるようになります。

システムの再起動後に、Windows のドライバー ウィザードによって自動的に新しい ioMemory デバイスが検出されて、ioMemory VSL の検索が開始される場合があります。この場合は、後の「インストール ウィザード」の手順までスキップできます。

1. [スタート]ボタン、[コントロール パネル]、[管理ツール]、[コンピュータの管理]、[デバイス マネージャ]の順にクリックします。
2. [Fusion ioMemory VSL devices]をクリックします。
3. 一覧内の ioMemory デバイスをクリックします。プロパティ ダイアログ ボックスが表示されます。

デバイスには `Mass Storage Controller` という名前が付いている場合があります。

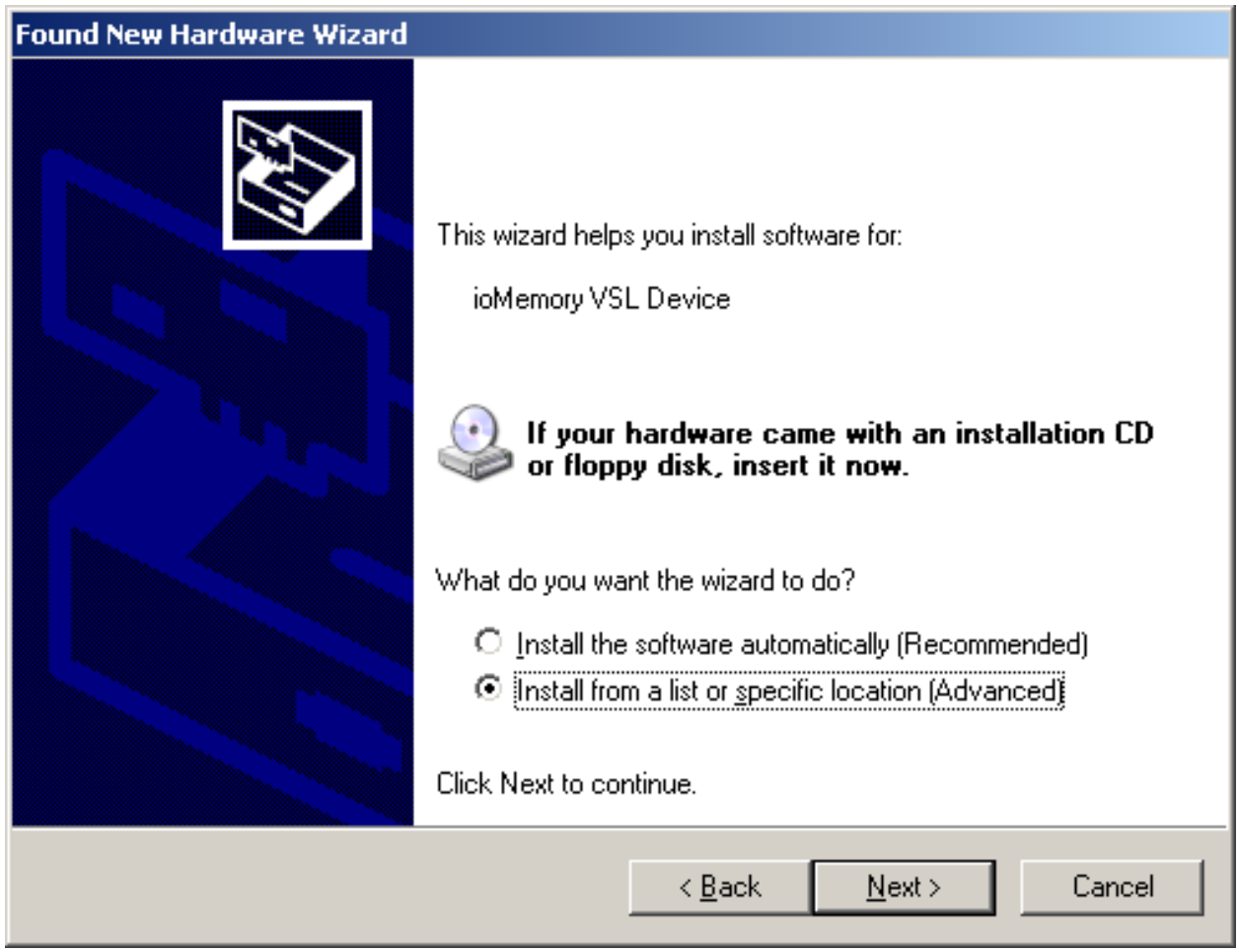


   a. [デバイスの状態]に「このデバイスは正常に動作しています。」と表示されている場合、ioMemory VSL がインストールされています。
   b. デバイスが正常に動作していない場合は、このデバイスに対して手動でソフトウェアをインストールする必要があります。引き続き手動インストールを行います。
4. プロパティ ダイアログ ボックスを閉じます。
5. デバイスを右クリックして、[ドライバの更新]を選択します。
6. 次に示す手順に従います。

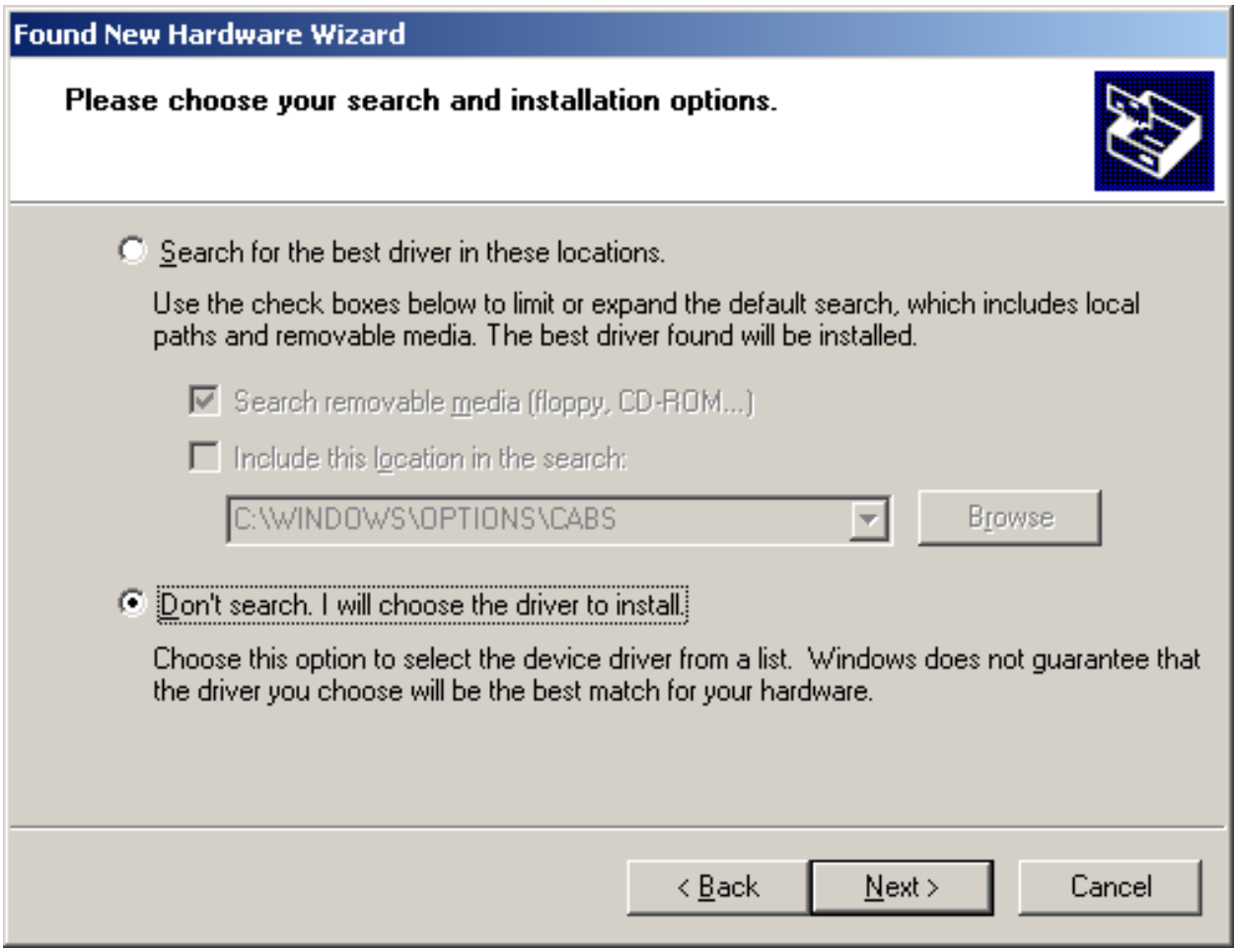
インストール ウィザード

1. Windows Update に接続してドライバーを検索するかどうかを尋ねるメッセージが表示されます。[いいえ、今回は接続しません]をクリックして、[次へ]をクリックします。

2. [一覧または特定の場所からインストールする]をクリックします。[次へ]をクリックします。

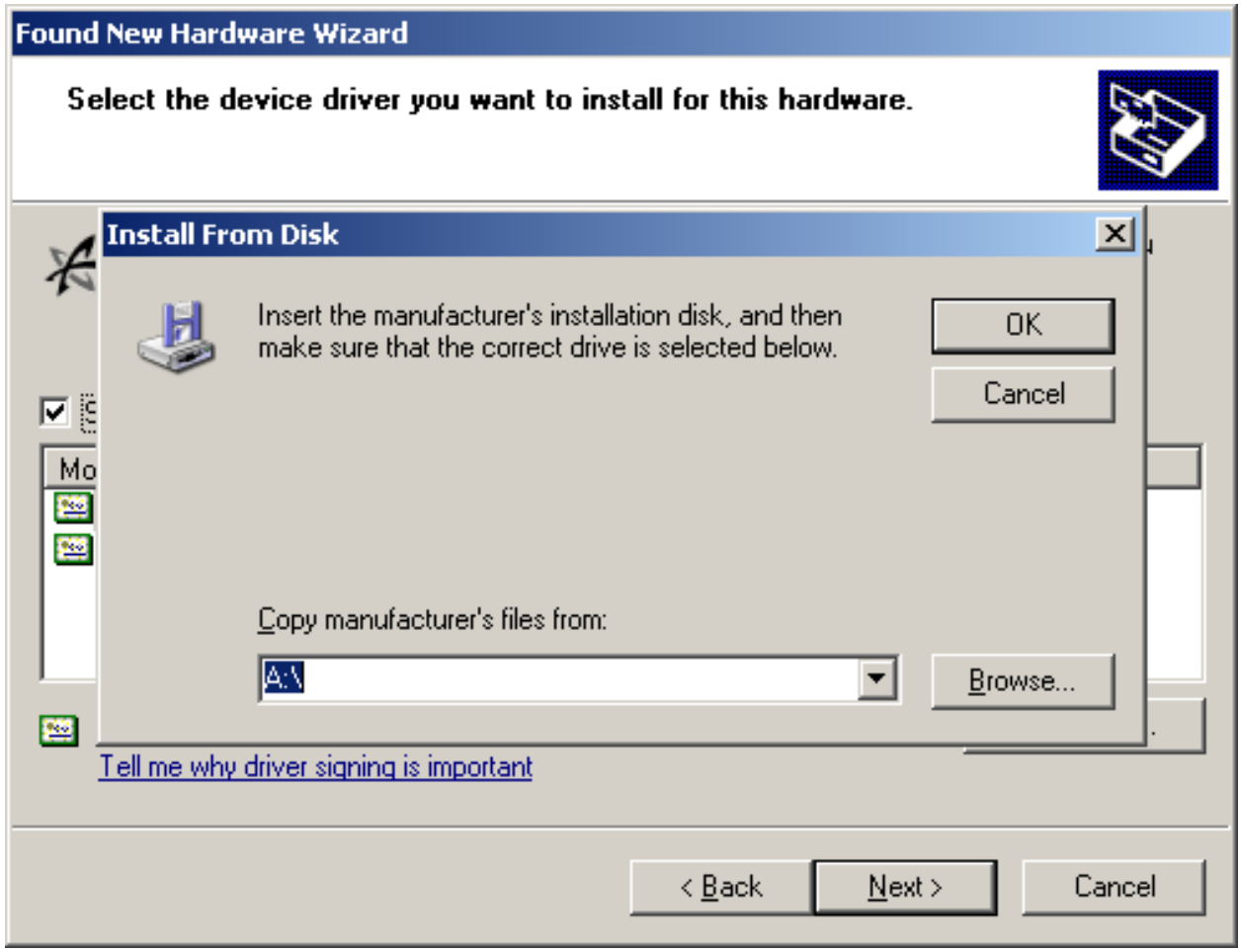


3. [検索しないで、インストールするドライバを選択する]をクリックします。[次へ]をクリックします。

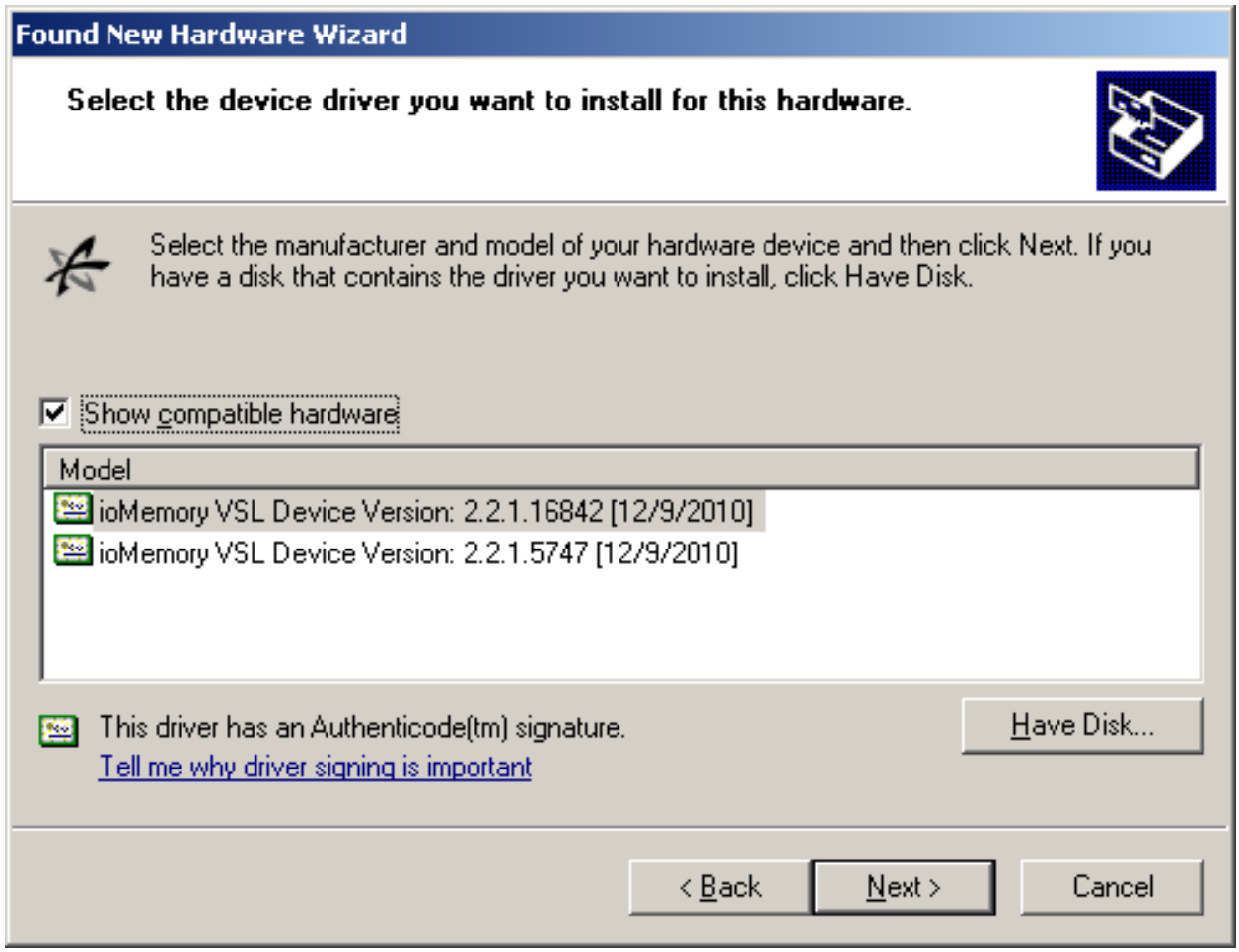


4. [ディスク使用]をクリックして、参照ダイアログ ボックスを表示します。

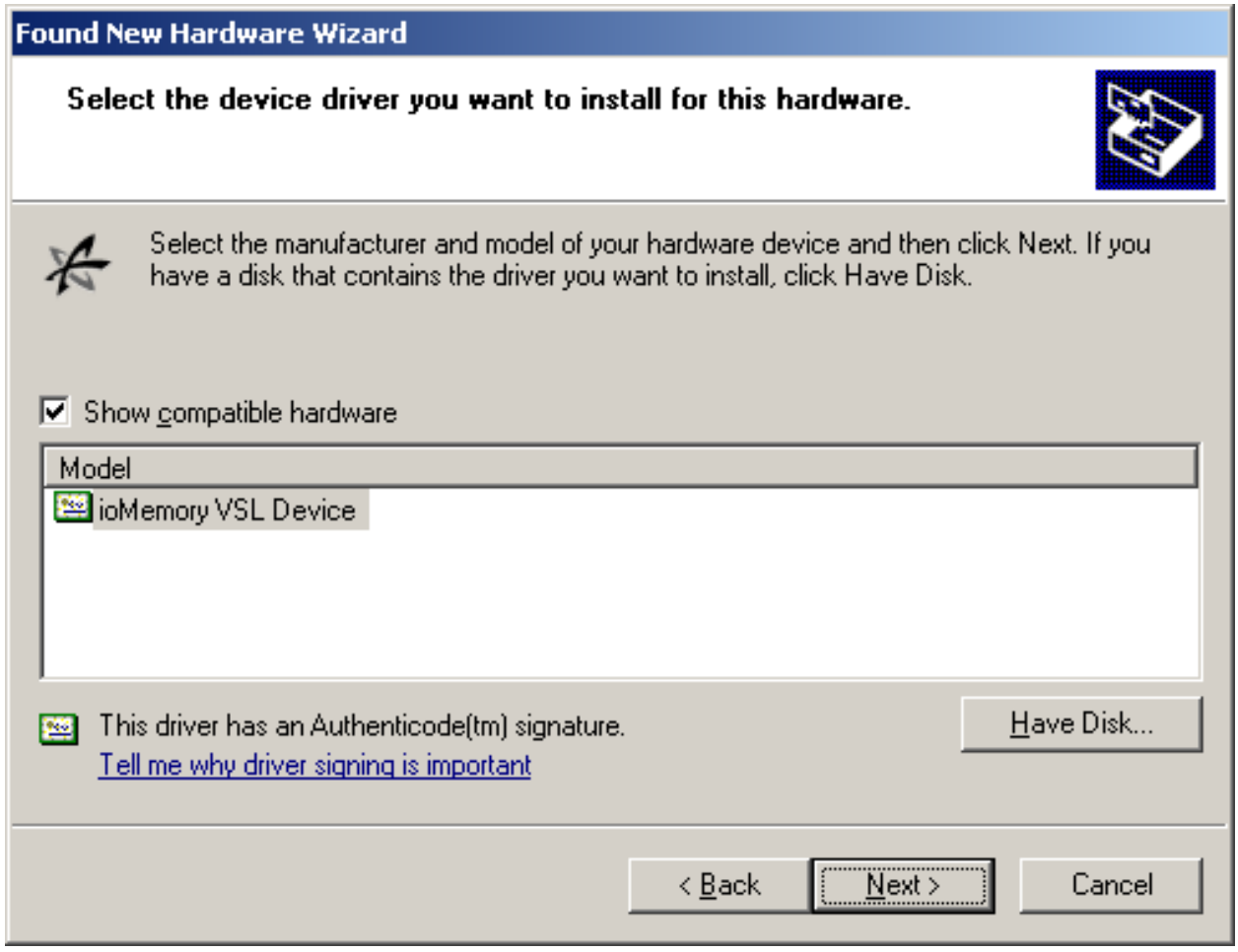
5. ioMemory VSL があるフォルダー（デフォルトは `C:\Program Files\Fusion-io ioMemory VSL\<VSL バージョン>\Driver`）を参照して選択します。



6. [OK]をクリックします。

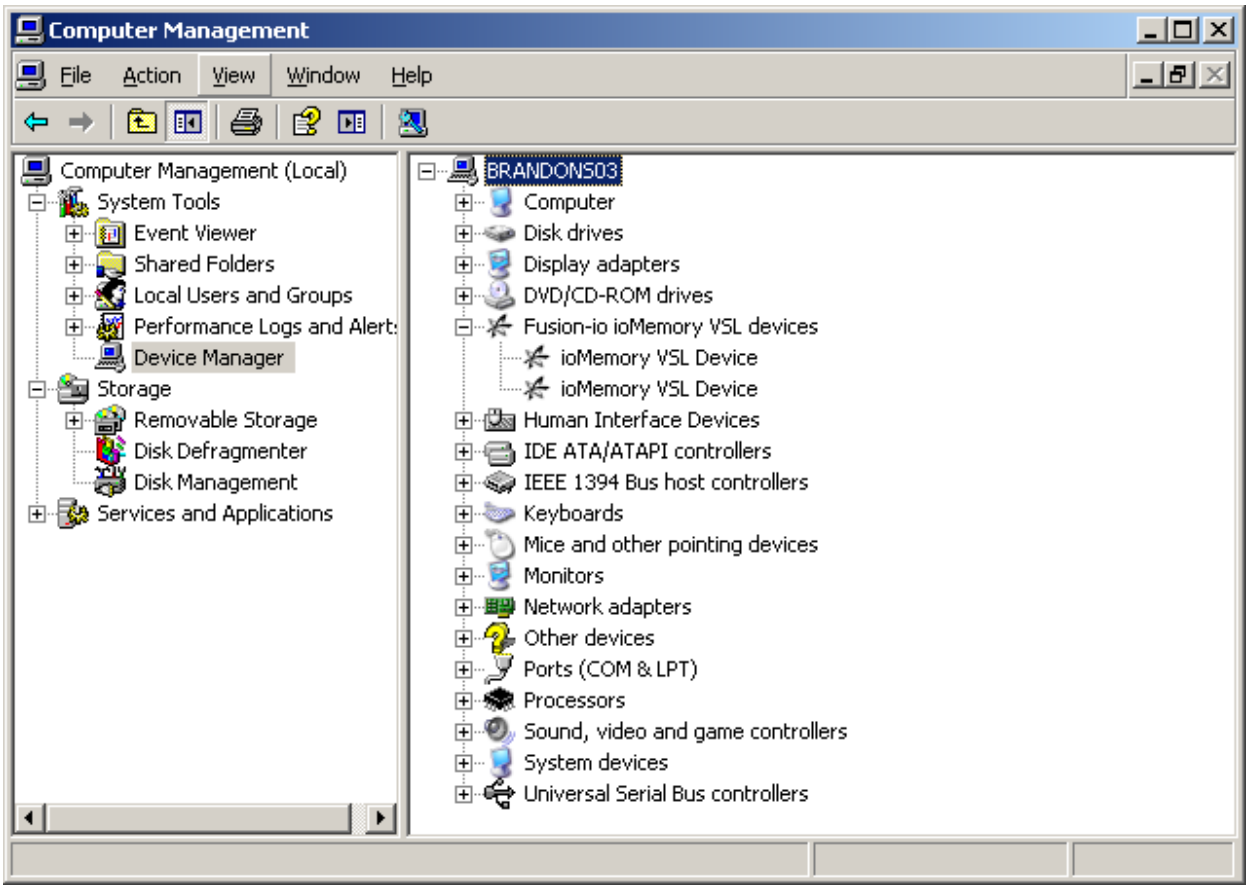


7. 選択したドライバーが、互換性のあるハードウェアのウィンドウに表示されます。このドライバーを選択して、[次へ]をクリックします。



ioMemory VSL のインストールが完了したら、次のメッセージが表示されます。

ioMemory VSL がデバイス マネージャーに表示されます。



ファームウェアを更新する必要がある場合は、このガイドの「古いファームウェアのチェック」および「デバイスのファームウェアのアップグレード」を参照してください。

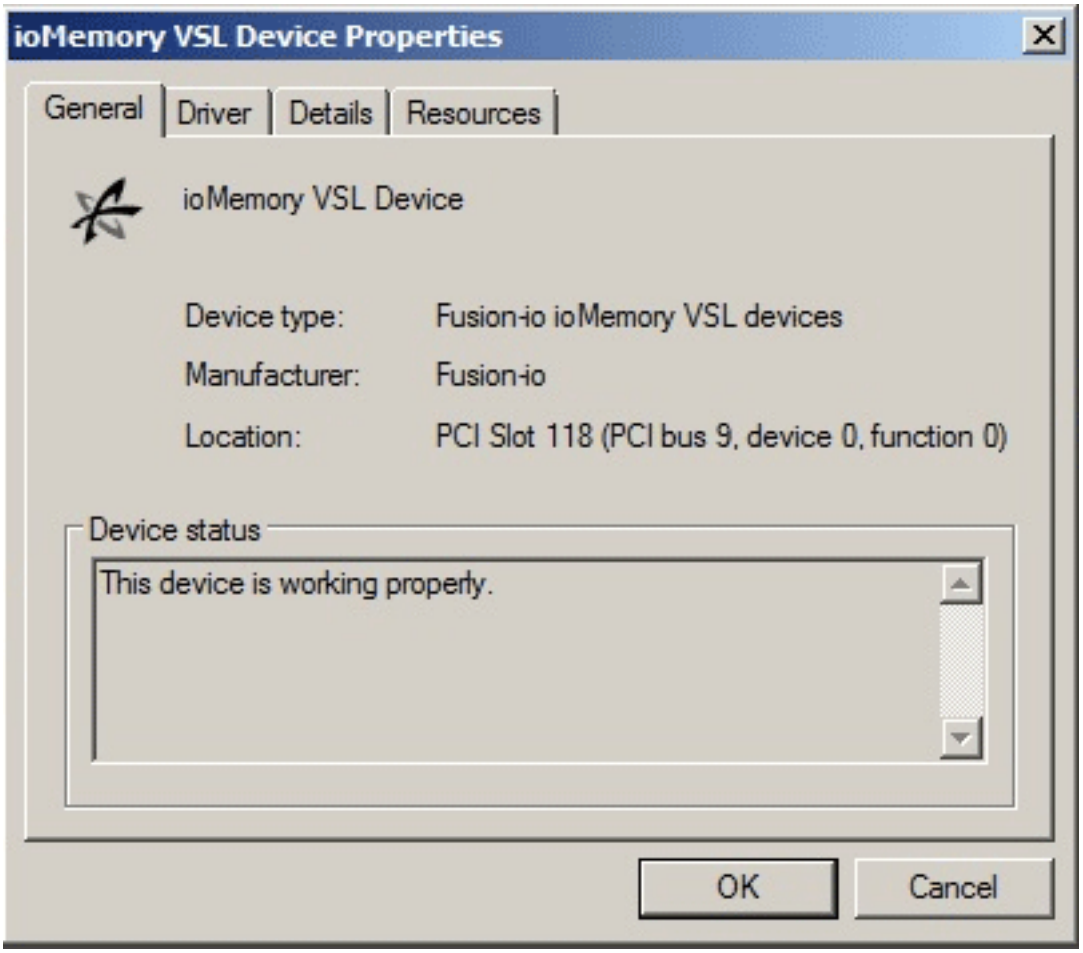
Windows Server 2008 での手動インストール

ioMemory VSL ドライバーを手動でインストールする前に、**PRIMERGY ダウンロードページ: http://jp.fujitsu.com/platform/server/primergy/downloads/**から ioMemory VSL の Windows 用セットアップ プログラムをダウンロードして、実行しておきます。これにより、システムに ioMemory VSL がインストールされ、各 ioMemory デバイスに ioMemory VSL をインストールできるようになります。

システムの再起動後に、Windows のドライバー ウィザードによって自動的に新しい ioMemory デバイスが検出されて、ioMemory VSL の検索が開始される場合があります。この場合は、後の「インストール ウィザード」の手順までスキップできます。

1. [スタート]ボタン、[コントロール パネル]、[デバイス マネージャ]の順にクリックします。
2. [Fusion ioMemory VSL devices]をクリックします。
3. 一覧内の ioMemory デバイスをクリックします。プロパティ ダイアログ ボックスが表示されます。

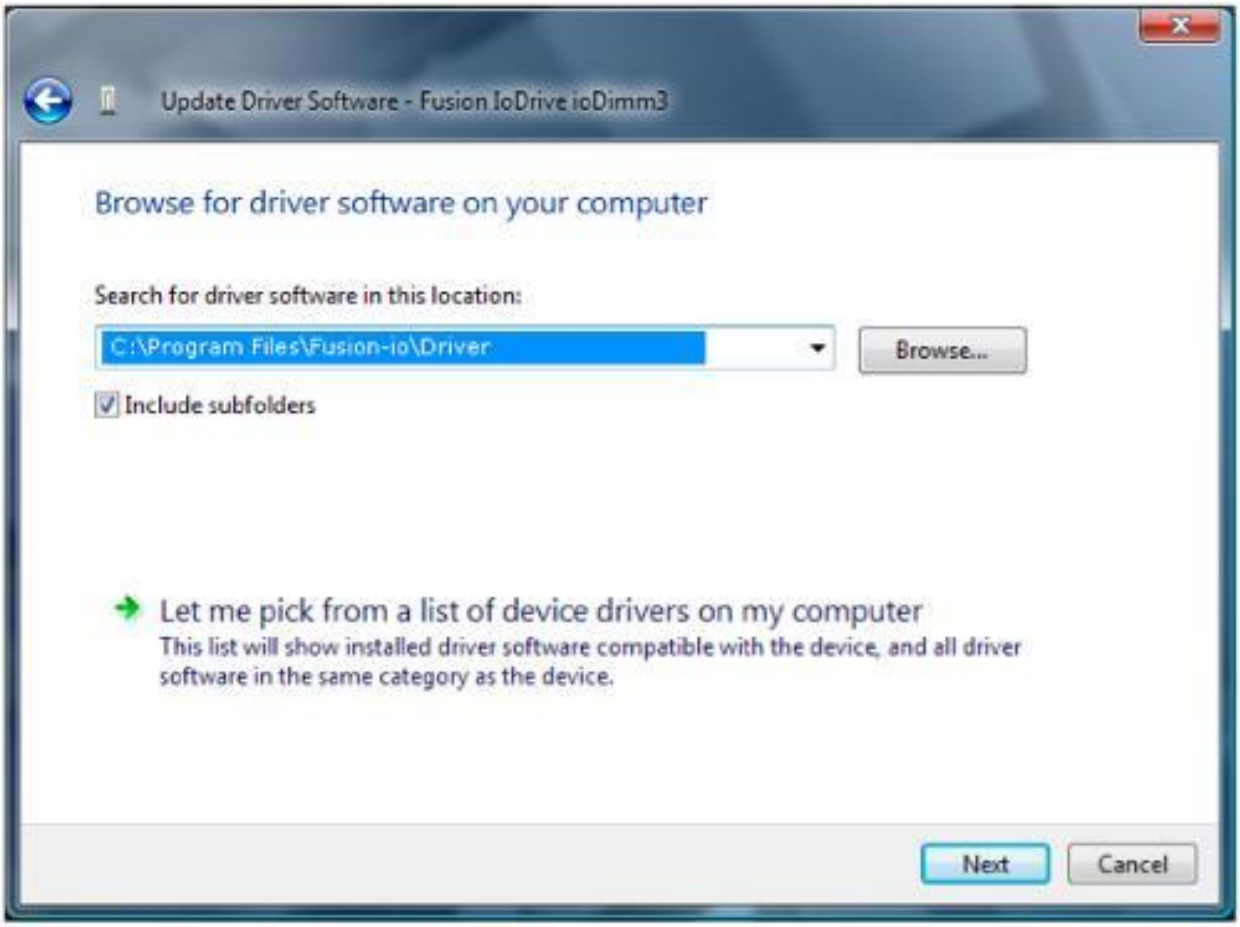
デバイスには `Mass Storage Controller` という名前が付いている場合があります。



   a. [デバイスの状態]に「このデバイスは正常に動作しています。」と表示されている場合、ioMemory VSL がインストールされています。
   b. デバイスが正常に動作していない場合は、このデバイスに対して手動でソフトウェアをインストールする必要があります。引き続き手動インストールを行います。
4. プロパティ ダイアログ ボックスを閉じます。
5. デバイスを右クリックして、[ドライバの更新]を選択します。
6. 次に示す手順に従います。

インストール ウィザード

1. ソフトウェア ドライバーを検索するように求めるメッセージが表示されます。



2. パス フィールドの横にある[参照]をクリックします。ファイル ダイアログ ボックスが表示されます。
3. ioMemory VSL があるフォルダー（デフォルトは `C:\Program Files\Fusion-io ioMemory VSL\<VSL バージョン>\Driver`）を選択します。
4. [OK]をクリックします。
5. [次へ]をクリックします。
   Windows によって正しいドライバーが検索されて、デバイス ソフトウェアがインストールされます。
   ドライバーのインストールが完了します。
6. コンピューターを再起動します。
7. 続いて、「古いファームウェアのチェック」に進みます。

付録 C – コマンドライン ユーティリティ

Windows 用セットアップ パッケージによって、次の表に示すさまざまなユーティリティが `C:\Program Files\Common Files\VSL Utils` フォルダーにインストールされます。

`C:\Program Files\Common Files\VSL Utils` ディレクトリには、ここに示されていない追加のユーティリティもインストールされます。これらの追加のユーティリティは、他のユーティリティに依存するユーティリティ（メイン VSL ユーティリティによって使用されるユーティリティ）であり、カスタマー サポートによって指示された場合を除いて直接使用しないでください。

ユーティリティ	目的
`fio-attach`	ioMemory デバイスを OS で利用可能にします。
`fio-beacon`	ioMemory デバイスの外部 LED を点灯します。
`fio-bugreport`	問題のトラブルシューティングに使用するための詳細レポートを作成します。
`fio-config`	デバイスの動作のための構成パラメータを有効にします。
`fio-detach`	一時的に ioMemory デバイスを OS のアクセスから切断します。
`fio-format`	ioMemory デバイスのローレベル フォーマットの実行に使用します。
`fio-pci-check`	PCI バス ツリーのエラー、特に ioMemory デバイスをチェックします。
`fio-status`	デバイスの情報を表示します。
`fio-trim-config`	TRIM 機能を有効または無効にします。
`fio-update-iodrive`	ioMemory デバイスのファームウェアを更新します。
`fio-sure-erase`	デバイスからデータをクリアまたはパージします。

管理者権限: コマンドライン ユーティリティを Windows で実行するには、管理者権限が必要です（メニュー項目の[コマンド プロンプト]を右クリックして、[管理者として実行]をクリックします）。
コマンド ラインからこれらのユーティリティを実行するには、コマンドがあるディレクトリ（デフォルトで `C:\Program Files\Common Files\VSL Utils`）に移動するか、またはシステム パスにこのディレクトリを追加する必要があります。Windows インストーラーを使用した場合は、ユーティリティをすぐに使用できるように、ユーティリティのディレクトリがシステム パスに自動的に追加されています。Windows インストーラーを使用していない場合は、使用している Windows のバージョンのドキュメントを参照して、システム パスにディレクトリを追加します。
どのユーティリティでも、`-h` オプション（ヘルプ表示）と`-v` オプション（バージョン表示）を使用できます。`-h` および`-v` を使用すると、情報の表示後にユーティリティが終了します。

fio-attach

説明

ioMemory デバイスを接続して、Windows で利用できるようにします。その後、Windows のディスクの管理ユーティリティを使用して、ioMemory デバイスをパーティション分割したり、RAID アレイの一部としてセットアップしたりできます。このコマンドを実行すると、接続プロセスの進行状況を示すプログレス バーおよびパーセンテージが表示されます。

ほとんどの場合、ioMemory VSL は、読み込み時に自動的にデバイスを接続して、スキャンを実行します。fio-attach は、fio-detach を実行した場合や、ioMemory VSL の AUTO_ATTACH パラメータを 0 に設定した場合にのみ実行する必要があります。
ioMemory デバイスが最小モード（minimal mode）の場合は、デバイスが最小モードとなった原因が修正されるまでの間、自動接続（auto-attach）は無効になります。

構文

```
fio-attach <デバイス> [オプション]
```

<デバイス>には、ioMemory VSL によってデバイスに設定された名前を指定します。この名前は `/dev/fctx` で、x は ioMemory デバイスを取り付けた PCIe バス番号を示しています（たとえば、`/dev/fct4` という名前は、Windows システムの PCIe バス 4 に取り付けられた ioMemory デバイスを示しています。このバス番号を表示するには、ioManager または `fio-status` を使用します）。

複数の ioMemory デバイスを指定できます。たとえば、`/dev/fct1 /dev/fct2` は、Windows システムの PCIe バス 1 および 2 に取り付けられた ioMemory デバイスを示しています。

オプション	説明
-c	正常にシャットダウンされている場合にのみ接続します。
-q	表示抑制: プログレス バーおよびパーセンテージの表示を無効にします。

fio-beacon

説明

デバイスの位置を識別するために ioMemory デバイスの 3 つの LED を点灯します。最初に ioMemory デバイスを切断してから、fio-beacon を実行する必要があります。

構文

```
fio-beacon <デバイス> [オプション]
```

<デバイス>には、ioMemory VSL によってデバイスに設定された名前を指定します。この名前は /dev/fctx で、x はデバイスの番号を示しています。

オプション	説明
-0	オフ: (ゼロ)3 つの LED を消灯します。
-1	オン: 3 つの LED を点灯します。
-p	<デバイス>に指定したデバイスの PCI バス ID を標準出力に出力します。使用方法およびエラー情報は、標準エラー出力ではなく標準出力に出力される場合があります。

fio-bugreport

説明

問題のトラブルシューティングに使用するためのデバイスの詳細レポートを作成します。

構文

```
fio-bugreport
```

注記

このユーティリティは、デバイスの現在の状態をキャプチャします。デバイスで性能または安定性に問題が発生している場合は、fio-bugreport ユーティリティを実行し、その出力を保存できます。

fio-bugreport では、複数の情報収集ユーティリティが実行され、結果として出力されたデータはそれぞれ 1 つのテキスト ファイルにまとめられます。結果は、utils ディレクトリ(デフォルトのインストール

パスは C:\Program Files\Common Files\VSL Utils)の下に、ユーティリティが実行された日時を示す名前を持つ.cab ファイルとして保存されます。

サンプル出力

```
C:\Users\username>"\Program Files\Fusion-io\Utils\fio-bugreport.exe"
Generating bug report. Please wait, this may take a while...
-----------------------------------------------
Gathering all Windows Event Logs...DONE
Gathering Fusion-io Windows Event Logs...DONE
Gathering System Information...DONE
Running fio utilities...DONE
Compressing to CAB file...DONE
Bug report has successfully been created:
fio-bugreport-20100222_192621.cab.
Please e-mail this file to support@fusionio.com
```

たとえば、fiobugreport-20090921.192621.cab という名前のバグ レポート ファイルのファイル名は、次の内容を示しています。

- 日付(20090921)
- 時刻(192621、つまり 19 時 26 分 21 秒)

fio-config

説明

デバイスの動作のための ioMemory VSL 構成パラメータを設定および取得します。パラメータの一覧については、後の「パラメータ リファレンス」を参照してください。

パラメータ値を適用するには、システムを再起動するか、またはデバイス マネージャーですべての ioMemory デバイスを無効にしてから再度有効にします。これにより、値が有効な状態で ioMemory VSL が再読み込みされます。再起動する場合は、-p オプションを指定してください。

構文

```
fio-config [オプション] [<パラメータ>] [<値>]
```

<パラメータ>は設定する ioMemory VSL パラメータ、<値>はパラメータに設定する値です。

オプション	説明

-e	構成パラメータの名前と値を列挙します。
-g <名前>	構成パラメータを取得します。
-p <名前>	構成パラメータを設定し、そのパラメータを永続的に保持します。**再起動後もパラメータ設定を維持する場合は、このオプションを指定します。**
-s <名前>	構成パラメータをメモリー内でのみ設定します。
-V	詳細情報を出力します。
-v	バージョン情報を出力します。

パラメータ リファレンス

次の表では、fio-config ユーティリティを使用して設定できる ioMemory VSL パラメータについて説明します。

fio-config にオプションとして指定するパラメータは、大文字で入力する必要があります。

MSI(メッセージ シグナル割り込み)はこのプラットフォームではデフォルトで有効になり、fio-config を使用して無効にすることはできません。

FIO_PREALLOCATE_MEMORY および FIO_EXTERNAL_POWER_OVERRIDE を除き、fio-config にオプションとして指定するすべてのパラメータはグローバルです。つまり、コンピューター内のすべての Fusion-io デバイスに適用されます。

FIO_PREALLOCATE_MEMORY および FIO_EXTERNAL_POWER_OVERRIDE パラメータを設定すると、**以前の値が上書きされます**。リストにシリアル番号を追加する場合は、新しいシリアル番号と以前に入力した番号の両方をリストに含める必要があります。リストをクリアするには、値を指定しないでパラメータを設定します。

オプション	デフォルト(最小/最大)	説明

AUTO_ATTACH	1 (0, 1)	ドライバーの読み込み時に常にデバイスを接続します(1)。
IODRIVE_TINTR_HW_WAIT	0 (0, 255)	ハードウェア割り込み間の待機間隔をマイクロ秒で指定します。
FIO_EXTERNAL_POWER_OVERRIDE	デバイスは非選択	選択されたデバイスにおいて、PCIe スロットからすべての電力を供給することを許可します。このパラメータの<値>は、アダプターのシリアル番号をコンマで区切ったリストです。 慎重に指定してください。詳細については、「PCIe Power Override の有効化」を参照してください。
FORCE_MINIMAL_MODE	0 (0, 1)	デバイスを強制的に最小モード(minimal mode)にします(1)。このパラメータは、デフォルトで false (0)に設定されています。
PARALLEL_ATTACH	0 (0, 1)	複数デバイスの並行接続を有効にします(1)。このパラメータは、デフォルトで false(0)に設定されています。
FIO_PREALLOCATE_MEMORY	0	選択されたデバイスに対して、ドライブをスワップ領域として使用可能にするために必要なすべてのメモリーを事前に割り当てます。次に例を示します。 `fio-config /dev/fct0 -p FIO_PREALLOCATE_MEMORY = "1234,54321"` 「1234」および「54321」は、fio-status によって取得したシリアル番号です。
WIN_DISABLE_ALL_AFFINITY	0(アフィニティは有効)	WIN_DISABLE_ALL_AFFINITY が 0 に設定されている場合、ドライバーでは、ドライバーに設定されている割り込みおよびワーカー スレッドのアフィニティが有効化されます。 WIN_DISABLE_ALL_AFFINITY が 1 に設定されている場合、ドライバーでは、すべてのアフィニティ

		の設定が無効化されます。この設定は、他のすべてのアフィニティの設定よりも優先されます。このパラメータが有効になるためには、ドライバーを再読み込みする必要があります。
WIN_DISABLE_DEFAULT_NUMA_AFFINITY	0（「デフォルトの」NUMA アフィニティが有効）	WIN_DISABLE_DEFAULT_NUMA_AFFINITY が 0 に設定されている場合、ドライバーは、初期化中に、OS がアダプターに割り当てたアフィニティの設定を Windows に問い合わせます。これは、「デフォルトの NUMA アフィニティ」と呼ばれています。アフィニティの問い合わせが正常に終了すると、ドライバーは、アダプターの割り込みおよび関連するワーカー スレッドのアフィニティをデフォルトの OS 設定に設定します。通常、これにより、割り込みおよびワーカー スレッドのアフィニティが、システム内の単一の NUMA ノードのすべてのプロセッサーに設定されます。 WIN_DISABLE_DEFAULT_NUMA_AFFINITY が 1 に設定されている場合、ドライバーは OS がアダプターに割り当てたアフィニティの設定を無視します。このパラメータが有効になるためには、ドライバーを再読み込みする必要があります。
FIO_AFFINITY	なし	FIO_AFFINITY は、システム内のすべてのアダプターのアフィニティ設定を指定する、3 つの項目が一組になった<アフィニティの指定>のリストです。3 つの項目のそれぞれはコンマで区切り、3 つの項目の各組はセミコロンで区切ります。 このパラメータの構文の詳細および使用例については、「付録 I – NUMA 構成」を参照してください。

fio-detach

説明

ioMemory デバイスを切断し、対応する `fctx` ブロック デバイスを OS から削除します。`fio-detach` ユーティリティは、デバイスでのすべての読み込み/書き込みアクティビティが完了するのを待ってから切断操作を実行します。このコマンドを実行すると、デフォルトで、切断の進行状況を示すプログレス バーおよびパーセンテージも表示されます。

このユーティリティを使用する前に、切断するデバイスが現在マウントされていないこと、および使用されていないことを確認してください。

構文

```
fio-detach <デバイス> [オプション]
```

<デバイス>には、ioMemory VSL によってデバイスに設定された名前を指定します。この名前は `/dev/fctx` で、x はデバイスの番号を示しています(番号は、ioDrive の PCIe バスを表しています)。たとえば、`/dev/fct4` という名前は、システムの PCIe バス 4 に取り付けられた ioMemory デバイスを示しています(この番号を表示するには、`fio-status` を使用します)。

複数の ioMemory デバイスを指定できます。たとえば、`/dev/fct1 /dev/fct2` は、Windows システムの PCIe バス 1 および 2 に取り付けられた ioDrive を示しています。

オプション	説明
`-f`	強制: 即時に切断を行います(メタデータは保存されません)。 `-f`(強制)オプションを使用すると、RAID セットアップ内の ioMemory デバイスであっても されますが、Windows のディスクの管理プラグインを使用してドライブまたはボリュームをオ インにしてから切断を実行することを強くお勧めします。強制的に切断を行うと、データが れる場合があります。
`-i`	即時: 即時に強制的に切断を行います(メタデータは保存されません)。デバイスが OS によって使用中の場合、この操作は失敗します。
`-q`	表示抑制: プログレス バーおよびパーセンテージの表示を無効にします。

注記

デバイスが使用中であるというエラーが表示されて、ioMemory デバイスの切断に失敗する場合があります。このエラーは、通常、デバイスがソフトウェア RAID(0、1、5)ボリュームの一部として使用されている場合に発生します。ioMemory デバイスが RAID ボリュームの一部として使用されており、切断するとボリュームが機能しなくなる可能性があるため、Windows は ioMemory デバイスに関連するドライブの切断要求を拒否します。単純なボリューム(単一の ioMemory デバイスのみを使用している場合など)

ではこのように拒否されることはありません。このようなエラーが発生した場合は、ディスクの管理 MMC プラグイン アプリケーションを使用してボリュームをオフラインにしてから切断します。

fio-format

説明

ioMemory デバイスは、出荷時にあらかじめフォーマットされています。そのため、デバイスの論理サイズやブロック サイズを変更する場合や、デバイス上のユーザー データを消去する場合を除き、通常は `fio-format` を実行する必要がありません。

デバイスのローレベル フォーマットを実行します。`fio-format` を実行すると、デフォルトで、フォーマットの進行状況を示すプログレス バーおよびパーセンテージによるインジケーターが表示されます。

このユーティリティを実行するとデバイス上のすべてのユーザー情報が削除されるので、慎重に使用してください。

4096 バイトなどの大きなブロック(セクター)サイズを使用すると、ioMemory VSL ホストのメモリー消費が最も多くなる場合のメモリー使用量を大幅に削減できますが、一部のアプリケーションは 512 バイト以外のセクター サイズと互換性がありません。

`-s` または`-o` オプションを指定しない場合、デバイスのサイズはデフォルトの公表されている容量に設定されます。`-s` および`-o` オプションを指定する場合は、サイズまたはパーセンテージも指定する必要があります。

構文

`fio-format [オプション] <デバイス>`

<デバイス>には、ioMemory VSL によってデバイスに設定された名前を指定します。この名前は `/dev/fctx` で、x は ioMemory デバイスを取り付けた PCIe バス番号を示しています(たとえば、`/dev/fct4` という名前は、Windows システムの PCIe バス 4 に取り付けられた ioMemory デバイスを示しています。このバス番号を表示するには、ioManager または `fio-status` を使用します)。

オプション	説明
`-b <サイズ B|K>`	ブロック(セクター)サイズをバイトまたはキロバイト単位で指定します(キロバイトの計算は 2 を底とする指数を使用します)。デフォルトは 512 バイトです。たとえば、`-b 512B` または`-b 4K` と指定します(`512B` における `B` の指定は任意です)。
`-f`	通常のチェックおよび警告をバイパスして、フォーマット サイズを強制的に適用します。このオプションは、`fio-format` が適切に進行しない例外的な状況で必要になる場合があります(`-y` オプションを指定しない場合は、「Are you sure?(実行してもよいですか?)」のプロンプトが表示されます)。
`-q`	表示抑制モード: プログレス バーおよびパーセンテージの表示を無効にします。

-s <サイズ M|G|T|%>	デバイスの容量を具体的なサイズ(TB、GB、MB 単位)または公表されている容量に対するパーセンテージで設定します。 • T: フォーマットするテラバイト(TB)数 • G: フォーマットするギガバイト(GB)数 • M: フォーマットするメガバイト(MB)数 • %: パーセンテージ(70%など。パーセント記号の指定が必要)
-o <サイズ B|K|M|G|T|%>	デバイス サイズを公表されている容量よりも大きいサイズにオーバーフォーマットします。最大サイズは、最大物理容量です。パーセンテージを使用する場合は、デバイスの最大物理容量に対する割合を指定します(-o オプションにはサイズを指定する必要があります。サイズの単位の説明については、上記の-s オプションを参照してください)。 このオプションを使用する前に、カスタマー サポートと具体的な使用例について相談してください。
-R	正常にシャットダウンされていない場合の高速再スキャンを無効にします。この場合、容量の一部が使用できなくなる可能性があります。
-y	アプリケーションからのすべての問い合わせに対して自動的に「yes(はい)」と回答します(プロンプトをバイパスします)。

fio-pci-check

説明

PCI バス ツリーのエラー、特に ioMemory デバイスをチェックします。このユーティリティは、各 ioMemory デバイスの現在のステータスを表示します。PCI-Express の標準エラー情報の出力、および状態のリセットも行います。

最初に fio-pci-check を実行したときにいくつかのエラー(通常は最大 5 つ程度のエラー)が表示されますが、問題ありません。正常に動作している場合、2 回目以降は、数時間の動作中に 1 つまたは 2 つのエラーのみが発生します。

このユーティリティを実行するには、ioMemory VSL が読み込まれている必要があります。一部の PCI エラーは、Windows ではリセットできません。

構文

fio-pci-check [オプション]

オプション	説明
-d <値>	1 = リンクを無効にします。0 = リンクをアップ状態にします。
-f	システム内のすべてのデバイスをスキャンします。

-r	リンクを維持されたままにします。
-v	詳細: ハードウェアについての追加データを出力します。
-y	続行するかどうかを尋ねられた場合に自動的に「yes(はい)」と回答します。

fio-status

説明

取り付けられている ioMemory デバイスについての詳細情報を提供します。このユーティリティは、fctx または fiox デバイスに対して動作します。このユーティリティは root として実行する必要があります。また、このユーティリティを実行する場合は、ioMemory VSL が読み込まれている必要があります。ioMemory VSL が読み込まれていない場合は、返されるステータス情報が少なくなります。

fio-status では、最小モード(minimal-mode)、読み取り専用モード(read-only mode)、書き込み低減モード(write-reduced mode)などの特定のエラー モードに対して、その状態が発生した原因を説明するアラートが表示されます。

構文

fio-status [<デバイス>] [オプション]

<デバイス>には、ioMemory VSL によってデバイスに設定された名前を指定します。この名前は /dev/fctx で、x はデバイスの番号を示しています(番号は、ioMemory デバイスの PCIe バスを表しています)。たとえば、/dev/fct4 という名前は、システムの PCIe バス 4 に取り付けられた ioMemory デバイスを示しています(fio-status ではこの番号が表示されます)。

fio-status では、<デバイス>が指定されていない場合、システム内のすべてのデバイスの情報が表示されます。ioMemory VSL が読み込まれていない場合、このパラメータは無視されます。

オプション	説明
-a	各デバイスで入手可能なすべての情報が報告されます。
-e	各デバイスのすべてのエラーおよび警告が表示されます。このオプションは問題の診断用であり、フォーマット サイズなどのその他の情報は表示されません。
-c	カウント: 取り付けられている ioMemory デバイスの数のみが報告されます。
-d	基本的な情報セット、および読み書きされたデータの合計量(累積データ量)が表示されます。-a オプションを指定した場合、このオプションは不要です。
-fj	JSON 形式: JSON 形式で出力を作成します。
-fx	XML 形式: XML 形式で出力を作成します。
-u	情報を入手できないフィールドを表示します。-fj または-fx と共に指定した場合にのみ有効です。

-U	情報を入手できないフィールド、およびその詳細な理由を表示します。-fj または-fx と共に指定した場合にのみ有効です。
-F<フィールド>	1 つのフィールドの値を出力します（フィールド名については次のオプションを参照）。デバイスを指定する必要があります。-F オプションは、複数指定できます。
-l	-F を使用して個別に指定できるフィールドを列挙します。
	出力の変更 バージョン 3.0.0 以降、fio-status の出力の標準フォーマットが変更されました。この変更によって、このユーティリティの出力を使用していたすべてのカスタム管理ツールに影響があります。

基本情報: オプションを指定しない場合、fio-status では、次の基本情報が報告されます。

- システムに取り付けられているカードの数およびタイプ
- ソフトウェア バージョン

アダプター情報:

- アダプター タイプ
- 製品番号
- 外部電源
- PCI 電力制限のしきい値（入手可能な場合）
- 接続されている ioMemory デバイス

ブロック デバイス情報:

- 接続ステータス
- 製品名
- 製品番号
- シリアル番号
- PCIe スロット番号
- ファームウェア バージョン
- デバイスのサイズ、および合計容量
- 平均温度（ioMemory VSL の読み込み以降の平均および最大、摂氏）
- ヘルスステータス: 正常（healthy）、摩耗が限界に近付いている（nearing wearout）、書き込み低減（write-reduced）、読み取り専用（read-only）
- 予約容量（パーセンテージ）
- 警告容量しきい値（パーセンテージ）

データ ボリューム情報: -d オプションを指定した場合は、基本情報に加えて次のデータ ボリューム情報が報告されます。

- 書き込まれた物理バイト数
- 読み込まれた物理バイト数

すべての情報: `-a` オプションを指定した場合は、すべての情報（基本情報、データ ボリューム情報、および次に示す情報）が出力されます。

アダプター情報:

- 製造業者番号
- 製造日付
- 停電時の保護ステータス
- PCIe バスの電圧（平均、最小、最大）
- PCIe バスの電流量（平均、最大）
- PCIe 電力制限のしきい値（ワット）
- PCIe スロットで利用可能な電力（ワット）
- PCIe でネゴシエートされたリンク情報（レーンおよびスループット）

ブロック デバイス情報:

- 部品番号
- 製造業者のコード
- 製造日付
- ベンダーおよびサブベンダー情報
- デバイスのサイズ、および合計容量
- フォーマット ステータスおよびセクター情報（デバイスが接続されている場合）
- FPGA ID およびローレベル フォーマット GUID
- PCIe スロットで利用可能な電力
- PCIe でネゴシエートされたリンク情報
- カードの温度（摂氏）
- 内部電圧: 平均および最大
- 補助電圧: 平均および最大
- 正常なブロック、データ、およびメタデータのパーセンテージ
- 累積データ ボリューム統計

エラー モード情報: ioMemory VSL が最小モード（minimal mode）、読み取り専用モード（read-only mode）、または書き込み低減モード（write-reduced mode）の場合に `fio-status` を実行すると、これらのモードではない場合と比較して、出力に次の相違点があります。

- 接続ステータスは「Status unknown: Driver is in MINIMAL MODE:（ステータス不明: ドライバーは最小モードです:）」などと表示されます。

- 最小モード(minimal mode)などの状態になっている理由が表示されます(「Firmware is out of date. Update firmware.(古いファームウェアが使用されています。ファームウェアを更新してください。)」など)。
- 「Geometry and capacity information not available.(形状および容量の情報は利用できません。)」と表示されます。
- メディアの正常性についての情報は表示されません。

fio-sure-erase

ベスト プラクティスとして、クリアまたはパージしたくない ioMemory デバイスがシステムに取り付けられている場合は、このユーティリティを使用しないことをお勧めします。最初に、誤って消去したくないすべてのデバイスを取り外してください。このユーティリティを使用してデータを削除すると、データは完全に削除されます。**復元することはできません。**

このユーティリティを使用する前に、保護する必要があるすべてのデータをバックアップしてください。

`fio-sure-erase` を使用した後、再度デバイスを使用する前に、fio-format を使用してデバイスをフォーマットします。

デバイスが読み取り専用モード(read-only mode)の場合は、`fio-sure-erase` を実行する前に `fio-format` を使用してフォーマットを実行します。デバイスが最小モード(minimal mode)の場合は、`fio-sure-erase` を使用してデバイスを消去することはできません。ファームウェアを更新すると、デバイスの最小モード(minimal mode)が解除される可能性があります

`fio-sure-erase` を実行するためには、ブロック デバイスを**切断しておく必要があります**。詳細については、「`fio-detach`」を参照してください。

説明

`fio-sure-erase` は、ioMemory デバイスから確実にデータを削除するコマンドライン ユーティリティです。このユーティリティは、次の標準規格における「Clear(クリア)」および「Purge(パージ)」レベルの破棄に準拠しています。

1. DOD 5220.22-M – フラッシュ EPROM に関する指示に準拠
2. NIST SP800-88 – フラッシュ EPROM に関する指示に準拠

クリアおよびパージのサポートの詳細については、後の説明を参照してください。

レジストリ要件

Windows では、ドライバーで ECC バイパス モードを構成するためのレジストリ キーを作成する必要があります。

1. 次のキーを検索します。

   ```
   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\fiodrive\Parameter
   s
   ```

2. このキーの下に、「BypassECC」という DWORD のキーを作成して、値を「1」に設定します。
3. コンピューターを再起動してからユーティリティを実行します。

構文

```
fio-sure-erase [オプション] <デバイス>
```

<デバイス>には、ioMemory VSL によってデバイスに設定された名前を指定します。この名前は `/dev/fctx` です。たとえば、`/dev/fct4` という名前は、Windows システムの PCIe バス 4 に取り付けられた ioMemory デバイスを示しています。このバス番号を表示するには、`fio-status` を使用します。

複数のデバイスを備えた製品

`fio-sure-erase` は、個別の ioMemory デバイスに対して動作します。たとえば、ioDrive Duo デバイスのパージを計画している場合は、この操作を ioDrive Duo デバイスの 2 つの ioMemory デバイスそれぞれに対して実行する必要があります。

オプション	説明
`-p`	クリアではなくパージ: 書き込み、消去の順で実行します。パージの詳細については、後の説明を参照してください。 パージする必要があるデバイスのサイズによっては、デバイスのパージが完了するまでに数時間かかることがあります。
`-y`	確認なし: ユーティリティの実行時に「yes(はい)」または「no(いいえ)」を入力する必要がありません。
`-q`	表示抑制: ステータス バーを表示しません。

オプションを指定しないで `fio-sure-erase` を実行すると、クリアが実行されます。詳細については、後の説明を参照してください。

メモリーの各ブロックは、1 または 0 のビットで均一に埋められます。

クリアのサポート

「クリア」は、`fio-sure-erase` を(オプションを指定しないで)実行するデフォルトの実行状態であり、リタイアしたイレースブロックを含む NAND メディア全体に対して完全なローレベル消去(すべてのセルの「1」への書き換え)を実行します。

オペレーションに必要なメタデータ(メディア イベント ログ、消去回数、読み書きされた物理バイト数、性能および温度履歴)は破棄されませんが、ユーザー固有のメタデータは破棄されます。

次に、クリア操作で実行される手順について説明します。

1. すべてのアドレス指定可能なブロックの統一マップが作成されます（これにより、`fio-sure-erase` で、以前にマッピングが解除された不良ブロックを含むすべてのブロックをアドレス指定できます）。
2. 各ブロックに対して、消去サイクルが実行されます（すべてのセルが「1」に書き換えられます）。
3. 不良ブロック マップが復元されます。
4. このユーティリティでは、クリア時にすべてのヘッダーが消去されるので、再度使用可能にするためにデバイスがフォーマットされます。

パージのサポート

「パージ」は、`fio-sure-erase` に`-p` オプションを指定することによって実行します。パージでは、最初に（リタイアしたイレースブロックを含む）NAND メディア全体が 1 つの文字で上書きされて（すべてのセルに論理表現の「0」が書き込まれます）、その後（リタイアしたイレースブロックを含む）メディア全体に対して全チップの消去（すべてのセルの「1」への書き換え）が実行されます。

オペレーションに必要なメタデータ（メディア イベント ログ、消去回数、読み書きされた物理バイト数、性能および温度履歴）は破棄**されませんが**、ユーザー固有のメタデータは破棄されます。

次に、パージ操作で実行される手順について説明します。

1. すべてのアドレス指定可能なブロックの統一マップが作成されます（これにより、`fio-sure-erase` で、以前にマッピングが解除された不良ブロックを含むすべてのブロックをアドレス指定できます）。
2. 各ブロックに対して、書き込みサイクルが実行されます（すべてのセルが「0」に書き換えられます）。
3. 各ブロックに対して、消去サイクルが実行されます（すべてのセルが「1」に書き換えられます）。
4. 不良ブロック マップが復元されます。
5. このユーティリティでは、クリア時にすべてのヘッダーが消去されるので、再度使用可能にするためにドライブがフォーマットされます。

fio-trim-config

説明

ファイル システムの利用可能な領域を再利用する TRIM 機能（Windows サービス）を有効または無効にします。一般的に、最大の性能を発揮するために TRIM を有効にすることをお勧めします。デフォルトで、TRIM は有効になっています。オプションを指定しないでこのユーティリティを実行すると、現在の TRIM ステータスが表示されます。

このユーティリティは、システム内のすべての ioMemory デバイスに適用されます。選択したデバイスに対してのみ TRIM を有効化または無効化することはできません。設定はすぐに反映されます。システムやデバイスを再起動する必要はありません。

構文

```
fio-trim-config [オプション]
```

オプション	説明
-d	このコンピューターで TRIM を無効にします。
-e	(デフォルト)このコンピューターで TRIM を有効にします。

fio-update-iodrive

アップグレードを行う前に、念のため ioMemory デバイスのデータをバックアップする必要があります。

説明

ioMemory デバイスのファームウェアを更新します。このユーティリティでは、すべての ioMemory デバイスを検出するために PCIe バスがスキャンされて、検出されたデバイスが更新されます。各デバイスに対して更新の進行状況を示すプログレス バーおよびパーセンテージが表示されます。

デバイスで障害が発生する可能性があるため、ファームウェアのアップグレード中は、絶対に電源をオフにしないでください。UPS が設置されていない場合は、ファームウェアのアップグレードを行う前にシステムに UPS を追加することを検討してください。

ファームウェアのアップグレードを複数回連続して実行する場合は、ファームウェアを 1 回アップグレードするごとにドライバーを読み込んでください。ドライバーを読み込まないと、ドライブ上のフォーマットが変更されず、データが失われます。

このユーティリティを使用して、ioMemory デバイスを以前のバージョンのファームウェアにダウングレードしないでください。ダウングレードを行うと、データが失われる危険性があり、また保証が無効となります。

(-d オプションを指定しない)デフォルトの処理では、<iodrive バージョン.fff>ファイルに含まれているファームウェアを使用して、すべての ioMemory デバイスがアップグレードされます。更新の実行前に、すべてのデバイスでアップグレードが必要であることを確認してください。確信を持てない場合は、-p(疑似実行)オプションを指定して、更新の推定結果を表示します。

ファームウェアを更新する前に、すべての ioMemory デバイスを切断する必要があります。

アップグレード パス

ioMemory デバイスのアップグレード時には、所定のアップグレード パスがあります。ioMemory デバイスをアップグレードする前に、このリリースの ioMemory VSL のリリース ノートを参照してください。

ファームウェアの更新時に midprom 情報の更新を求めるエラー メッセージが表示された場合は、カスタマー サポートにご連絡ください。

構文

`fio-update-iodrive [オプション] <iodrive バージョン.fff>`

<iodrive バージョン.fff>は、パスおよび Fusion-io が提供するファームウェア アーカイブ ファイルです。デフォルトのパスは C:¥Program Files¥Fusion-io ioMemory VSL¥Firmware です。このパラメータは必須です。

オプション	説明
`-d`	指定したデバイスを更新します。デバイスは `fctx` の形式で指定します。x は、`fio-status` で表示されるデバイスの番号です。このオプションを指定しない場合は、すべてのデバイスが更新されます。 誤った ioMemory デバイスを更新するとデバイスが損傷する危険性があるため、`-d` オプションは慎重に指定してください。
`-f`	強制的にアップグレードを実行します（主に、以前のファームウェア バージョンへのダウングレードに使用されます）。ioMemory VSL が読み込まれていない場合は、このオプションに加えて`-d` オプションも指定する必要があります。 デバイスが損傷する危険性があるため、-f オプションは慎重に指定してください。
`-l`	アーカイブ内で利用可能なファームウェアを列挙します。
`-c`	デバイスに設定されたロックをクリアします。
`-p`	疑似実行: 推定される更新内容を表示します。ただし、実際のファームウェアは変更されません。
`-q`	プログレス バーやパーセンテージを表示せずに更新プロセスを実行します。
`-y`	すべての警告メッセージに対して自動的に「yes（はい）」と回答します。

更新プロセス中は、3 つすべての外部 LED インジケーターが点灯します。

付録 D – TRIM のサポート

はじめに

TRIM は、ソリッドステート デバイスに特有の問題に対処するために使用されます。この問題は、ユーザーがファイルを削除した場合に、そのファイルが含まれているデバイスで、削除された領域が再利用可能であると認識されないことが原因で発生します。デバイスでは、削除されたデータを上書きできるという通知をシステムから受信するまでの間、そのデータが引き続き有効であるとみなされます。通常のハード ディスク デバイスでは、大きな性能低下を引き起こすことなくデバイスの同じセクターに対して継続的に書き込みを行えるため、このような問題は発生しません。一方、ソリッドステート デバイスでは、同じ論理セクターへの書き込みを行っても、異なるフラッシュ メモリー領域にデータが書き込まれます。

フラッシュ メモリーではこのように書き込みが行われるため、全体が SSD で構成されている場合には、書き込みスループットが大幅に低下する可能性があります。現在利用可能なストレージ スタックおよびファイル システムは、この注意点を考慮した設計になっていません。SSD ソフトウェアでは、すべてのセクターに有効なデータが含まれているとみなさざるを得ず、そのためデバイスは常にいっぱいであるものとして扱われます。実際には、通常、ファイル システムにデバイスの全容量に相当するデータが含まれていることはありません。

TRIM は既存のファイル システムに対する拡張機能であり、有効なユーザー データが含まれていない論理セクターを SSD ソフトウェアに通知します。TRIM は、ファイルが削除されたときにファイル システムからこの情報を取得して、デバイスに対して、このファイルが占有していた領域を再利用できることを通知します。ウェアレベリング ソフトウェアは、この機能によって、将来の書き込み操作の処理用にその領域を確保しておき、後で再利用できます。

この機能を実行するために、TRIM はバックグラウンドで実行され、Fusion-io デバイスで使用されているファイル システムを継続的に監視します。このプロセスは、コンピューターの性能に大きな影響を与えないように、あえて遅延的に実行されます。

プラットフォーム

TRIM 機能は、Windows Server 2003 以降の Windows オペレーティング システムで使用される ioMemory デバイスで利用できます。

TRIM 機能は、Windows 7 および Windows Server 2008 R2 オペレーティング システムに組み込まれています。ioMemory デバイスでは、Windows 7 の TRIM がサポートされています。つまり、Fusion-io の TRIM は Windows Server 2008 R2 では不要であり、このオペレーティング システムでは実行されません。システムの変更は不要です。Windows Server 2008 R2 オペレーティング シ

ステムが検出されると、Fusion-io の TRIM サービスは自動的に停止します。

TRIM サービスの使用

TRIM は、起動プロセスの速度を低下させないように、コンピューターの起動から数分後にバックグラウンドで Windows サービスとして自動的に実行されます。ただし、後述の構成可能な設定を使用して、TRIM を調整できます。

TRIM の開始と停止

TRIM サービスを Windows サービス マネージャーで開始または停止するには、次の手順に従います。

1. 「Fusion-io Trim Service」を検索します。
2. [開始]または[停止]をクリックします。

サービスを停止した場合、現在のセッションでのみ停止されます。コンピューターを再起動すると、TRIM が再度実行されます。

TRIM の有効化

Fusion-io の TRIM サービスを有効にするには、次の手順に従います。

1. レジストリ エディターを開きます。
2. 次のキーに移動します。

   ```
   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\fiodrive\Parameters
   ```

3. `DWORD` として `TrimEnabled` 値を追加します（後の説明を参照）。（上記の 2 つの方法のいずれかを使用して）TRIM サービスを停止し、再度サービスを開始するか、またはコンピューターを再起動します。

`fio-trim-config` ユーティリティを使用すると、`TrimEnabled` フラグが自動的に設定されます。変更内容を有効にするためにシステムを再起動したり、サービスを手動で再起動したりする必要はありません。レジストリをユーザーが編集すると予期しない結果を招くことがありますが、このユーティリティを使用するとレジストリの編集は必要ないため、このユーティリティを使用して TRIM を有効化および無効化することをお勧めします。

TrimEnabled

ユーザーが上記の 2 つの方法のいずれを使用して TRIM を開始および停止するかにかかわらず、このレジストリ値を指定すると、TRIM を実行するかどうかを制御できます。この値が存在し、1 に設定され

ている場合、TRIM は通常どおりに実行されます（上記の方法でサービスを停止していない場合）。Fusion-io の TRIM サービスの起動前にこの値が 0 に設定されている場合は、この値を 1 に変更して（またはレジストリから削除して）、サービスを手動で再起動するまで、サービスは実行されません、この値が存在しない場合は、ユーザーが停止しない限りサービスが実行されます。この値はセッションを超えて保持されます。そのため、この値を 0 に設定してコンピューターを再起動した場合、コンピューターが再度起動したときに TRIM は実行されません。

TRIM の即応性（アグレッシブネス）の制御

TRIM サービスの即応性を制御できます。TRIM のデフォルト設定では、ioMemory デバイスが最大の性能を発揮できるように、コンピューターの性能への影響が最小限に抑えられます。ただし、書き込み負荷の高いアプリケーションでドライブを使用している場合は、TRIM の即応性のレベルを上げることができます。これを行うには、レジストリのエントリをいくつか追加する必要があります。

1. レジストリ エディターを開きます。
2. 次のキーに移動します。

   ```
   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\fiodrive\Parameters
   ```

3. `DWORD` として `TrimTimeDelayInSeconds` 値を追加します（後の説明を参照）。
4. `DWORD` として `BitmapRetrievalDelayInMilliseconds` 値を追加します（後の説明を参照）。
5. （上記の 2 つの方法のいずれかを使用して）TRIM サービスを停止し、再度サービスを開始するか、またはコンピューターを再起動します。

TrimTimeDelayInSeconds

このレジストリ値によって、TRIM がシステム上のすべてのボリュームの処理を終了した後、どの程度の時間が経過した後に再度実行を開始するかが制御されます。たとえば、この値を 600（10 分を秒で表した数値）に設定すると、TRIM はシステム上のすべてのボリュームを処理し、10 分間休止して、再度ボリュームを処理するというサイクルを繰り返します。この値には、0～4,294,967,295 の任意の数値を指定できます。この値が設定されていない場合や、レジストリに存在しない場合、TRIM ではデフォルトの 900 秒（15 分）が使用されます。

BitmapRetrievalDelayInMilliseconds

この値の効果は、CPU に対する TRIM の影響度に最もよく表れます。現在のデフォルト値は 25 ミリ秒ですが、0（即時に応答）～4,294,967,295（実用性なし）の任意の数値を指定できます。

構成

TRIM は、次の構成および機能で使用できます。

- RAID ボリューム（ミラー、スパン、またはストライプ）。注記: RAID 5（パリティ付きストライプ）は現在サポートされていません。
- シンプル ボリューム（RAID なし）。
- 上記 RAID レベルの複数デバイスにわたる任意の組み合わせ（RAID セット内に 1 つ以上の ioMemory デバイスがある場合）。
- 同じ ioMemory デバイス上の複数のパーティション。
- NTFS および FAT32 ファイル システム。
- マウント ポイントありのボリューム（<ドライブ文字>は未定義）。
- 圧縮ボリューム。
- 異なるクラスター サイズ、パケット サイズ、およびセクター サイズ。
- 拡張および縮小されたボリューム。

付録 E – ioMemory デバイスの正常性の監視

ここでは、データを保護し、デバイスの寿命を延ばすために ioMemory デバイスの正常性を測定および監視する方法について説明します。

NAND フラッシュおよびコンポーネントの障害

ioMemory デバイスは、コンポーネントの障害、およびソリッドステート ストレージに特有の書き込み回数による利用制限に対して多くのレベルでの保護を備えた、高度なフォールトトレランス性を持つサブシステムです。しかしながらどのようなストレージ サブシステムでも、コンポーネントの障害が発生する場合があります。

ホストされているアプリケーションに対して適切な NAND フラッシュ メディアを選択し、デバイスの累計読み書きバイト数および正常性を積極的に監視することによって、製品の設計寿命期間全体にわたって信頼できる性能を発揮できます。

正常性ステータスの指標

ioMemory VSL は、事前に定められた使用停止しきい値を使用して、ブロックの使用率を監視します。ioManager および `fio-status` ユーティリティでは、開始時の値が 100 で 0 までカウント ダウンされる正常性のインジケーターが表示されます。一定のしきい値を下回ると、各種の措置が取られます。

10%の正常性しきい値では、警告が一度表示されます。このアラーム イベントのキャプチャ方法については、後の「正常性（ヘルスモニターリング）の監視手法」を参照してください。

0%のしきい値では、デバイスが正常ではないとみなされます。デバイスは、書き込み低減モード（write-reduced mode）になります。これにより、寿命が少し伸びるため、その間にデータをデバイス外に安全に移行できます。この状態の ioMemory デバイスは、書き込み性能が低くなることを除いて正常に動作します。

0%のしきい値に達した後、デバイスはすぐに読み取り専用モード（read-only mode）になります。このモードでは、ioMemory デバイスへの書き込みを試みるとエラーが発生します。一部のファイル システムで読み取り専用のブロック デバイスをマウントするためには、読み取り専用でマウントすることの指定以外に、特別なマウント オプションが必要となる場合があります。

たとえば、Linux の ext3 ファイル システムでは、「`-o ro,noload`」を指定する必要があります。「`noload`」オプションを指定すると、ファイル システムに対してジャーナルの再生を行わないように通知されます。

読み取り専用モード(read-only mode)になった後にデバイスを引き続き使用すると、デバイスで障害が発生する可能性がより高くなるため、このモードはデバイスからデータを移行できる最後のチャンスとなります。

ioMemory デバイスは、障害モード(failure mode)になる場合があります。この場合、デバイスはオフラインになり、アクセスできなくなります。内部で回復不可能な障害が発生した場合、不適切な手順でファームウェアがアップグレードされた場合、デバイスが消耗した場合などにこのモードになります。

サービスまたは保証関連の質問については、デバイスを購入した販売会社にご連絡ください。

複数の ioMemory デバイスを備えた製品では、これらのモードは各デバイスに対して独立に管理されます。

正常性(ヘルスモニターリング)の監視手法

`fio-status`: `fio-status` ユーティリティの出力には、正常性のパーセンテージとデバイスの状態が表示されます。これらの項目は、次のサンプル出力では「`Media status`(メディア ステータス)」として表示されています。

```
Found 1 ioDrive in this system
Fusion-io driver version: 2.2.3 build 240
Adapter: ioDrive
Fusion-io ioDrive 160GB, Product Number:FS1-002-161-ES
...
Media status: Healthy; Reserves: 100.00%, warn at 10.00%; Data:  99.12%
Lifetime data volumes:
Physical bytes written:  6,423,563,326,064
Physical bytes read   :  5,509,006,756,312
```

ioManager: [Device Report]タブで、右側の列にある[Reserve Space]のパーセンテージを確認します。パーセンテージが高いほど、ドライブの正常性が高いことを示しています。

SNMP/SMI-S: Windows や Linux で SNMP または SMI-S の正常性インジケーターを構成する方法の詳細については、対応する付録を参照してください。

`fio-status` ユーティリティでは、次の正常性ステータス メッセージが出力されます。

- Healthy(正常)
- Low metadata(メタデータ低下)
- Read-only(読み取り専用)
- Reduced-write(書き込み低減)
- Unknown(不明)

ソフトウェア RAID と正常性監視

ソフトウェア RAID スタックは、通常、従来のストレージ メディアの障害モードを検出して、その影響を軽減できるように設計されています。ioMemory デバイスは、障害の影響を可能な限り軽減しようと試みますが、これらの新しい障害対応メカニズムは既存のソフトウェア RAID スタックと互換性があります。RAID グループ内の ioMemory デバイスでは、a）デバイスが書き込み低減（write-reduced）状態になり、b）書き込み作業負荷が高い場合に、十分な速度でデータを書き込むことができなくなります。この場合、デバイスは RAID グループから切り離されます。読み取り専用モード（read-only mode）のデバイスは、デバイスから書き込み I/O に失敗したというエラーが返されたときに切り離されます。回復不可能な障害は、従来のストレージ デバイスでの障害と同様に検出および処理されます。

付録 F – ioMemory デバイスでの Windows ページ ファイルの使用

はじめに

この付録では、Windows で ioMemory デバイス上のページング（スワップ）ファイルを効率的に使用する方法について説明します。

従来のディスク ドライブでページ ファイルを使用する場合は、RAM と比較してディスク ドライブの性能が劣るために、使用可能なページ ファイルおよび仮想メモリーのサイズに事実上の制限があります。OS のページング ファイルを 1 つ以上の ioMemory デバイスに配置すると、より大きなページ ファイルおよび仮想メモリーを使用できます。これは、ioMemory デバイスはハード ディスクと比較して、大幅に応答時間が短く、帯域幅が広いためです。

デバイスでのページングのサポートの構成

ioMemory VSL を構成して、1 つ以上の ioMemory デバイスでページング ファイルをサポートできます。このためには、ページング ファイルで使用する各 ioDrive で、考えられるすべての I/O シナリオにおいて必要になる可能性がある最大のメモリー量を事前に割り当てておく必要があります。このメモリーの割り当ては、アダプター（ioDIMM）インスタンスごとに行います。

ホストで追加の RAM メモリーが使用されるため、ページングは、ページング ファイルを実際に保持する ioMemory デバイスでのみ有効化する必要があります。複数の ioMemory デバイスにそれぞれ 1 つのページング ファイルを配置することもできます。この場合は、利用可能なすべてのページング ファイルにページング I/O がストライピングされるため、仮想メモリー（VM）サブシステムの性能が向上する可能性があります。

ioMemory VSL による RAM の消費

ioMemory デバイスごとに事前に割り当てられる RAM の量は、デバイス全体のサイズおよび（`fio-format` を使用して）ドライブをフォーマットするときに選択したセクター（ブロック）サイズに応じて異なります。

ioMemory デバイスの GB あたりの RAM 使用量については、このバージョンのソフトウェアのリリースノートを参照してください。

セクター サイズを大きくすると、ページングをサポートするために必要なホストのメモリー消費量を大幅に削減できます。a）4K はホストのメモリー ページの一般的なサイズであり、b）全体的なホストのメモリー消費量を最小限に抑えることができるため、4K のセクター サイズを使用することをお勧めします。Windows の NTFS では、通常、4K のクラスター サイズが使用されるため、512 バイトのセクター サイズとしか互換性がないアプリケーション（Windows XP や Windows 2003 など）を除いて、512 バイトでフォーマットしても有用ではありません。

示されているメモリー量は、ページングをサポートする ioMemory デバイスごとに必要になります。どの ioMemory デバイスをページング ファイルの保持に使用するかを慎重に計画する必要があります。

非ページ メモリー プール

ioMemory デバイスの事前割り当てされるメモリーは、Windows カーネルの非ページ メモリー プールから割り当てられます。このプールは、システム コンポーネントが追加のカーネル メモリーを消費すると動的にサイズが大きくなります。このプールの最大サイズには、次の制限があります。

- Windows Server 2003、2008 R1/R2 – RAM の 75%、最大 128 GB まで
- Windows Vista/Windows 7 – RAM の 40%、最大 128 GB まで

ページ ファイルの使用を計画する場合は、使用済みの非ページ プール メモリーの量に注意する必要があります。これは、ioMemory デバイスによって RAM が事前に割り当てられますが、これにより利用可能な物理非ページ メモリーの量が減るためです。事前割り当てメモリーの合計と、使用済みの非ページ メモリーを合わせたサイズが最大非ページ メモリー プールのサイズを超えた場合は、ioMemory VSL の読み込みに失敗します。

次に、2 つの ioMemory デバイスによる非ページ メモリー プールの合計使用量の例を示します。

- 1 つの ioMemory デバイスでは 850 MB の RAM を必要とし、もう 1 つの ioMemory デバイスでは 1,700 MB の RAM を必要とします。

  RAM 要件については、このバージョンの ioMemory VSL のリリース ノートを参照してください。

- どちらのデバイスも 4K セクター サイズでフォーマットされています。
- 両方のデバイスでページング ファイルをサポートします。

現在の割り当て済み非ページ プールの値はタスク マネージャーで確認できます。この例では、576 MiB です（タスク マネージャーの値は、MiB 単位で表示されます（1024 x 1024 = 1 MiB））。システム上の RAM の合計は 8,000 MB で、OS は Windows Server 2008 R2 です。

最初に、576 MiB を MB に変換します。`576 MiB * (1 MB/1.048576 MiB) = ~549 MB`

利用可能な非ページ プールの合計サイズを計算するには、次の式を使用します。

```
(8000 MB x 0.75) - 549 - 850 - 1700
```

非ページ プールには、2,901 MB の利用可能な領域が残っています。

ページングのサポートの有効化/無効化

メモリーの事前割り当ては、ioMemory VSL の初期化中に行われます。ページングのサポートを有効化するには、`FIO_PREALLOCATE_MEMORY` 構成項目を有効化する必要があります。この項目は、`fio-config` コマンドライン ユーティリティを使用して有効化できます。このパラメータには、ページング ファイルをサポートする ioMemory デバイスの 10 進表記のシリアル番号のリストを含む文字列を割り当てます。ioMemory VSL では、これらのインスタンスに対してメモリーの事前割り当てが実行されます。

次に、`fio-config` ユーティリティを使用して、シリアル番号 1234 および 17834 の 2 つの ioMemory デバイスに対してページングおよび事前割り当てを有効化する例を示します。シリアル番号情報は、`fio-status` ユーティリティを使用して取得できます。

```
fio-config -p FIO_PREALLOCATE_MEMORY "1234,17834"
```

すべてのデバイスでページングのサポートを無効化するには、`FIO_PREALLOCATE_MEMORY` に値 0 を指定します。

```
fio-config -p FIO_PREALLOCATE_MEMORY "0"
```

現在の値を問い合わせるには、次のコマンドを実行します。

```
fio-config -g FIO_PREALLOCATE_MEMORY
```

ioManager を使用して、ページングのサポートを管理（有効化または無効化）することもできます。

新しい事前割り当て設定を有効にするには、ioMemory VSL を再読み込みする必要があります。通常、ioMemory VSL の再読み込みを行うには、コンピューターを再起動するか、またはデバイス マネージャーで ioMemory デバイスの各インスタンスを無効にしてから有効にします。

また、Windows のシステムのプロパティを使用してページング ファイルの構成を変更する場合は、プロパティを適用するためにシステムを再起動する必要があります。したがって、FIO_PREALLOCATE_MEMORY とシステムのページ ファイル構成の両方を変更して、その後再起動を一度行うと、両方が適用されます。

Windows のページ ファイルの管理

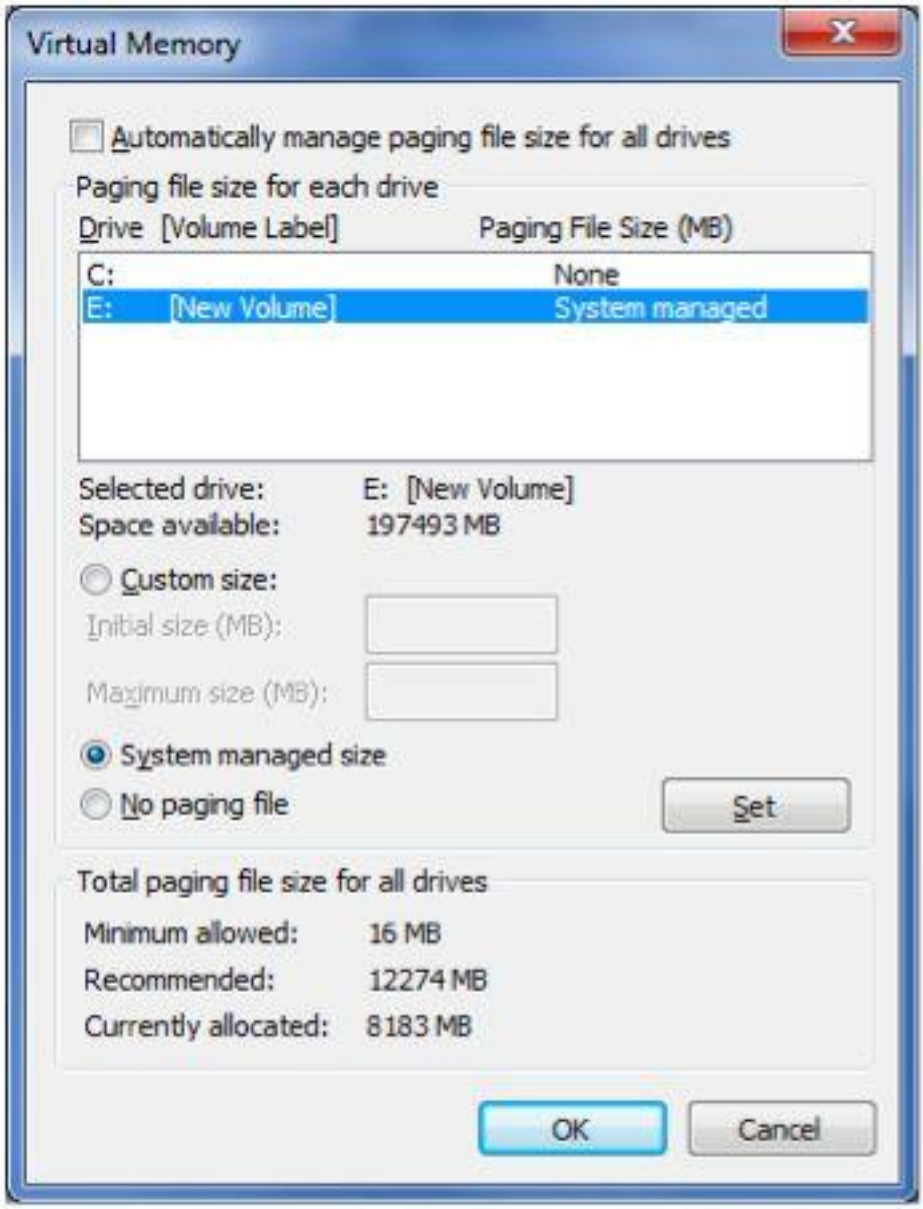
デフォルトで、ioMemory VSL では、ページ ファイルのサポートが無効になっています。前のセクションでは、1 つ以上の ioMemory デバイスでページ ファイルのサポートを有効化する方法について説明しました。ここでは、Windows に組み込まれているコントロール パネルを使用して、ioMemory デバイス上でページング ファイルを構成およびセットアップする方法について説明します。

ページング ファイルのセットアップ

Windows でページ ファイルをセットアップするには、次の手順に従います。

1. コントロール パネルに移動して、[システム]をダブルクリックします。
2. [システムのプロパティ]ウィンドウで[詳細設定]タブをクリックします。
3. [詳細設定]タブで、[パフォーマンス]の[設定]をクリックします。[パフォーマンス オプション] ダイアログ ボックスが表示されます。

4. [詳細設定]タブで、[変更]をクリックします。[仮想メモリ]ダイアログ ボックスが表示されます。



このダイアログ ボックスを使用して、システム内で利用可能な各ドライブに対してページ ファイルを構成できます。[すべてのドライブのページング ファイルのサイズを自動的に管理する]チェック ボックスをオンにすると、Windows によってシステム ドライブ(OS が起動するドライブ)にページ ファイルが 1 つ作成されます。ioMemory デバイスにページング ファイルを作成する場合は、このチェック ボックスをオフにする必要があります。

Windows では、最大 16 個の異なるページング ファイルがサポートされます。ioMemory デバイスでページ ファイルを有効にするには、次の手順に従います。

1. デバイスの一覧から、ioMemory デバイスを選択します。
2. [カスタム サイズ]をクリックします。
3. [初期サイズ]ボックスおよび[最大サイズ]ボックスに値を指定します。
4. [設定]をクリックして、設定内容を保存します。この手順は省略しないでください。省略すると、変更内容が失われます。
5. [OK]をクリックします。
6. 再起動を求めるメッセージが表示されたら、[はい]をクリックします。新しいページ ファイル設定を有効にするには、再起動が必要です。

ドライブ上のページング ファイルを削除するには、上記の手順に従い、[ページング ファイルなし]を選択します。通常は、性能上の理由により、システム ハード ディスク上のすべてのページング ファイルを削除します。

[仮想メモリ]ダイアログ ボックスでは、ioMemory デバイスでページ ファイルのサポートが構成されていない場合でも、利用可能な ioMemory デバイスに対してページ ファイルを構成できます。この場合、ダイアログ ボックスでページ ファイルを有効にすることはできますが、必要な再起動を行っても、デバイス上にページ ファイルは作成されません。このドキュメントで説明した指示に従って、1 つ以上の ioMemory デバイスでページ ファイルのサポートを適切に有効化してください。

システム ドライブでのページング ファイルの構成

Windows では、デフォルトで、システムの起動ドライブ（通常はハード ディスク）にページ ファイルが作成されて管理されます。通常は、この起動ドライブに Windows がインストールされます。ハード ディスクの I/O 性能は ioMemory デバイスと比較して大幅に劣るため、システム ハード ディスク上に通常のページ ファイルを保持することは一般的に最適とはいえません。この状態を改善するために、後で説明する手順に従って、システム起動ドライブのページ ファイルを削除したり、ページ ファイルのサイズを最小化したりできます。（システム ドライブではなく）ioMemory デバイスでページ ファイルを有効化すると、仮想メモリー（VM）マネージャーによって利用可能なすべてのページ ファイルの間で I/O がストライピングされるため、VM サブシステムの性能が高まります。また、ioMemory デバイスは非常に大きなメモリー ストアとして動作するため、大規模なアプリケーションにおけるメモリー使用状況を大幅に改善できます。

Windows カーネルでは、システム ディスクのページ ファイルを使用して、クラッシュ ダンプが保存されます。クラッシュ ダンプには、小さいもの（ミニダンプ）や大きいもの（カーネル全体のメモリー ダンプ）があります。通常はダンプ ファイルのサポートなし、または小さいダンプ ファイルに対応するだけで十分です。システム ドライブのページ ファイル構成には、いくつかの選択肢があります。

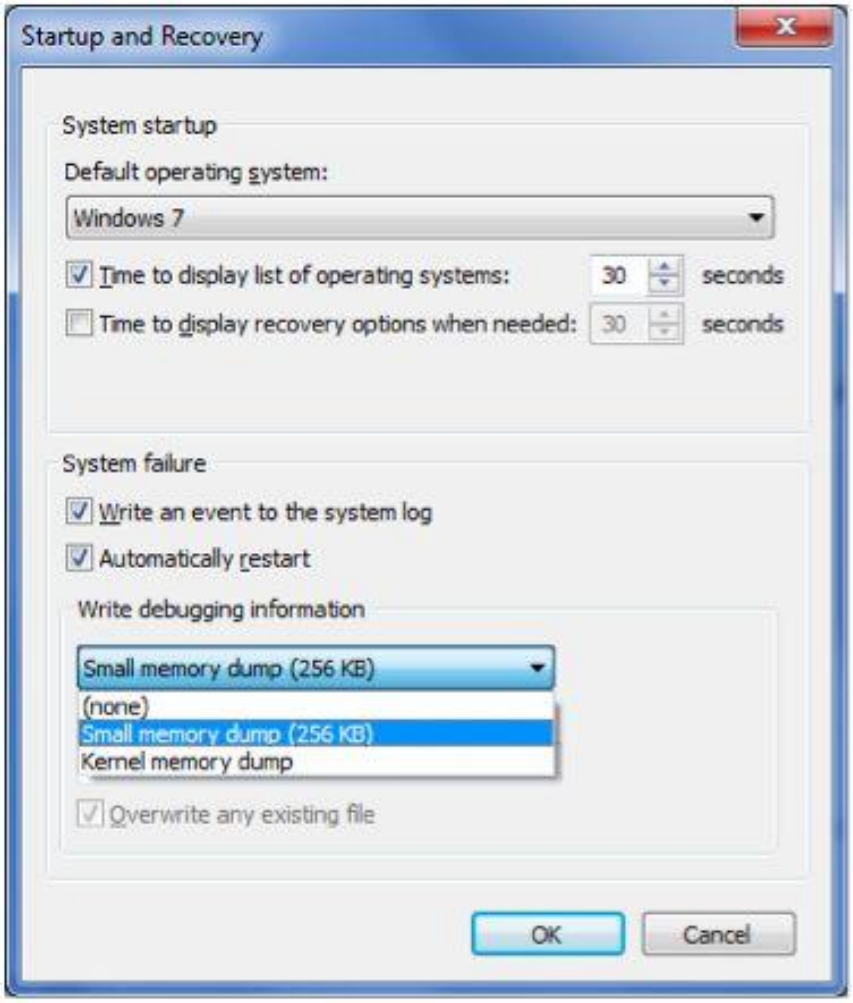
1. システム起動ドライブを含むすべてのハード ディスクのすべてのページ ファイルを削除します。これにより、ioMemory デバイス上でのページング I/O が最大化されますが、以後、システム クラッシュが発生した場合にクラッシュ ダンプ ファイルは作成されません。ただし、システム ドライブでページ ファイルを再度有効化して、クラッシュ シナリオを再現することができる場合があります。

2. システム起動ドライブに最小限のサイズのページ ファイルを作成します。推奨される最小サイズは 16 MB です。ただし、Windows によって、最低でも 400 MB のページ ファイルが必要であるという警告が表示される場合があります。
3. カーネル全体のメモリー ダンプに十分な大きさのページ ファイルを作成します。この場合は、通常、最低でも取り付けられている RAM のサイズと同じサイズのページ ファイルが必要です。RAM のサイズの 1.5 倍のサイズのページ ファイルを作成することをお勧めします。

Fusion-io では、ioMemory デバイスをクラッシュ ダンプの保存先とすることができるように、現在積極的に取り組んでいます。

クラッシュ ダンプの構成を表示または変更するには、次の手順に従います。

1. [システムのプロパティ]ダイアログ ボックスを表示します。
2. [詳細設定]タブをクリックします。
3. [起動と回復]の[設定]をクリックします。[起動と回復]ダイアログ ボックスが表示されます。



[システム エラー]で、システム ログの処理、再起動、およびデバッグ情報の設定を変更できます。

最小コミット可能メモリーの確保

[仮想メモリ]ダイアログ ボックスで[システム管理サイズ]を有効にするか、または[カスタム サイズ]を設定する場合は、慎重に設定してください。初期サイズがコミット可能仮想メモリーの必要量未満の場合、コミットされるメモリー量が現在割り当てられているページ ファイル サイズまたは初期サイズの値を超えると、アプリケーションでメモリー割り当てエラーが発生する可能性があります。コミットされるメモリーが現在のページ ファイル サイズを超えると、追加のメモリー割り当て要求に失敗します。Windows の仮想メモリー マネージャーは、ドライブの利用可能なサイズ、またはカスタム設定の「最大サイズ」のいずれか小さい方のサイズまで、ページング ファイルのサイズを徐々に増やします。

大量の仮想メモリー(RAM のサイズの 1.5 倍以上のメモリー)をコミットする場合にアプリケーションのメモリー割り当てエラーを回避するには、アプリケーションでコミットされる予想メモリー使用量に応じて、コミットされるメモリーの初期値および最大値を明示的に設定する必要があります。これらの値は、通常、同じ値に設定する必要があります。

ページング ファイルの必要な大きさ

ページ ファイルの適切なサイズの決定方法の詳細については、次の記事を参照してください。

- Microsoft の関連記事のリンク: Windows の限界に挑む
- 仮想メモリーに限定して説明した記事: Windows の限界に挑む: 仮想メモリ

ページ ファイルの動作の確認

ページ ファイルが ioMemory デバイス上に配置され、有効になっていることを確認するには、ドライブのルートで隠しファイルを検索します。たとえば、プロンプトで次のコマンドを実行します。

```
dir c: /ah
```

出力の一覧に、`pagefile.sys` というファイルが表示されます。ページ ファイルが存在しない場合は、[仮想メモリ]ダイアログ ボックスでページ ファイル構成を再チェックして、対象の ioMemory デバイスでページ ファイルのサポートが有効になっていることを確認します。

性能

ioMemory デバイスをページング ストアとして使用すると、仮想メモリー システムの全体的な性能を高めることができます。実際にどの程度のメリットがあるかは、アプリケーションによる仮想メモリー使用状況、およびハードウェア プラットフォームとその性能によって大きく異なります。

付録 G – SNMP のテスト モードおよび MIB のサポート

はじめに

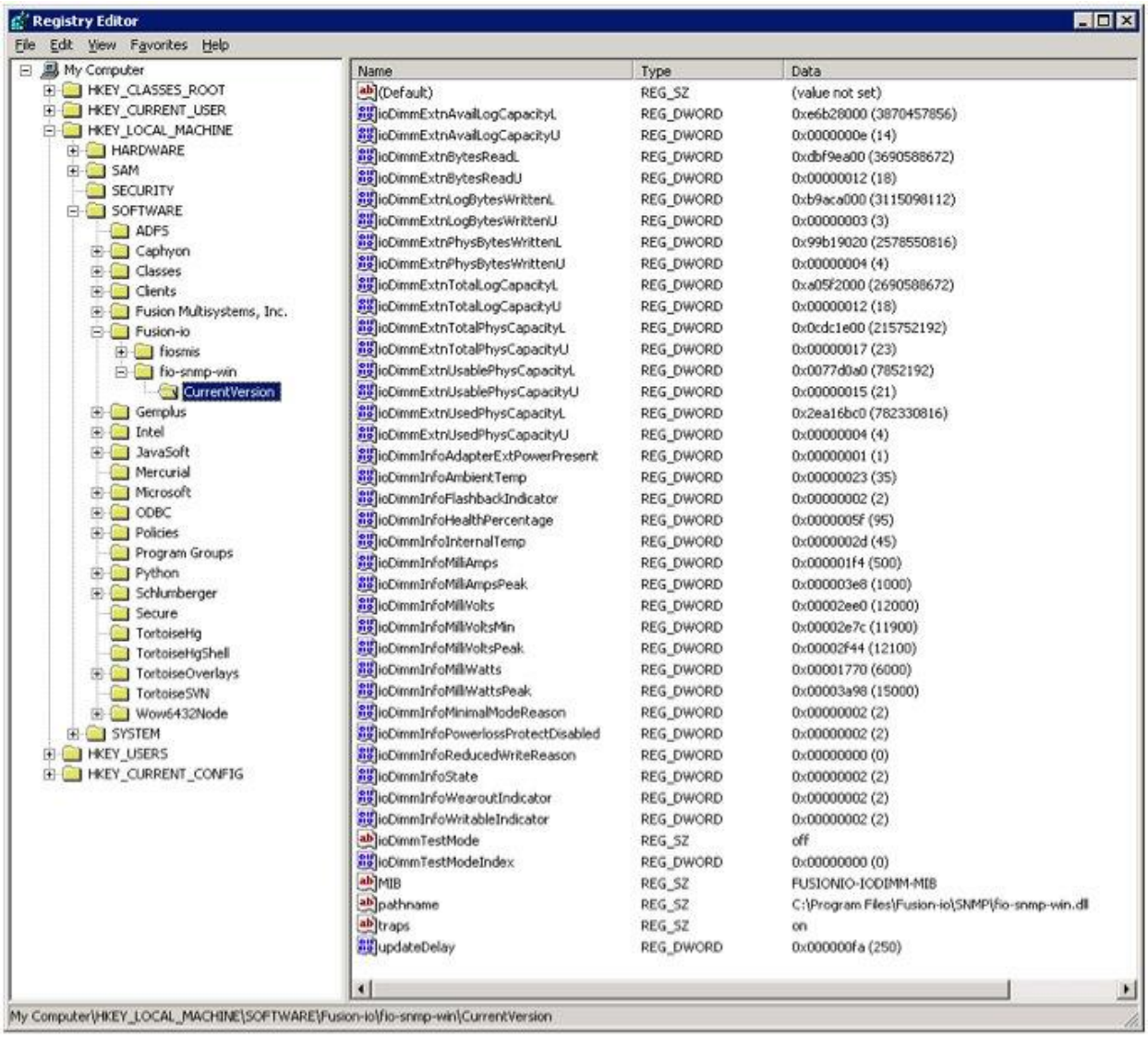
ここでは、VSL Windows SNMP エージェントでテスト モードをセットアップする方法について説明します。テスト モードでは、Windows レジストリにテスト値を設定して、デバイスで実際の状態を発生させることなく SNMP トラップを強制的に送信できます。

たとえば、SNMP テスト モードを使用し、書き込み不能インジケーターを変更してトラップを生成したり、デバイスの物理サイズまたは論理サイズの変更をシミュレートしたりできます。

SNMP テスト モードを使用するには、ioMemory VSL で SNMP オプションをインストールしておく必要があります。

テスト モードのレジストリ値の使用

下図は、SNMP テスト値用に用意されたレジストリ エントリを示しています。



次の表では、これらの各エントリについて説明します。アスタリスク(*)が付いているエントリに指定された値を設定すると、SNMP トラップが生成されます。この変更によって、`fusionIoDimmMIBCondition MIB` 変数および `fusionIoDimmInfoStatus MIB` 変数が影響を受ける可能性があります。

	が付いているエントリを除き、すべてのエントリでは、レジストリの変更がすぐに反映されます。が付いているエントリの変更内容は、Windows SNMP エージェントを再起動しないと反映されません。

SNMP テスト レジストリ エントリ	説明

ioDimmExtnAvailLogCapacityL	利用可能な論理容量(バイト単位)の下位ワード
ioDimmExtnAvailLogCapacityU	利用可能な論理容量(バイト単位)の上位ワード
ioDimmExtnBytesReadL	デバイスがフォーマットされて以降読み込まれた合計バイト数の下位ワード
ioDimmExtnBytesReadU	デバイスがフォーマットされて以降読み込まれた合計バイト数の上位ワード
ioDimmExtnLogBytesWrittenL	書き込まれたユーザー データのバイト数の下位ワード
ioDimmExtnLogBytesWrittenU	書き込まれたユーザー データのバイト数の上位ワード
ioDimmExtnPhysBytesWrittenL	書き込まれた合計物理バイト数の下位ワード
ioDimmExtnPhysBytesWrittenU	書き込まれた合計物理バイト数の上位ワード
ioDimmExtnTotalLogCapacityL	フォーマットされた合計論理容量(バイト単位)の下位ワード
ioDimmExtnTotalLogCapacityU	フォーマットされた合計論理容量(バイト単位)の上位ワード
ioDimmExtnTotalPhysCapacityL	フォーマットされた合計物理容量(バイト単位)の下位ワード
ioDimmExtnTotalPhysCapacityU	フォーマットされた合計物理容量(バイト単位)の上位ワード
ioDimmExtnUsablePhysCapacityL	利用可能な物理容量(バイト単位)の下位ワード。これは、有効なデータを保持している領域、消去されて書き込み可能状態になっている領域、またはガベージ コレクションによって再利用可能になるのを待機している領域の容量です。
ioDimmExtnUsablePhysCapacityU	利用可能な物理容量(バイト単位)の上位ワード。これは、有効なデータを保持している領域、消去されて書き込み可能状態になっている領域、またはガベージ コレクションによって再利用可能になるのを待機している領域の容量です。
*ioDimmInfoInternalTemp	デバイスの現在の内部温度(摂氏)。ioDimm カードに対してこの値を 78℃より高い温度に設定すると、トラップが生成されます。HP メザニン カードでは、90℃より高い温度に設定すると、トラップが生成されます。
*ioDimmInfoFlashbackIndicator	1 = Flashback の冗長性機能が低下している、2 = 低下していない
*ioDimmInfoNonWritableIndicator	1 = 読み取り専用(read-only)のしきい値を超えたため、デバイスに書き込むことができなくなった、2 = 書き込み可能
ioDimmInfoPercentLifeRemaining	デバイスの残り寿命値
*ioDimmInfoState (trap generated if state = 4) (状態が 4 の場合にトラップ生成)	Current state of the attached client device(接続されたクライアント デバイスの現在の状態): unknown(不明)(0) detached(切断済み)(1) attached(接続済み)(2)

	minimal（最小）（3） error（エラー）（4） detaching（切断中）（5） attaching（接続中）（6） scanning（スキャン中）（7） formatting（フォーマット中）（8） updating（更新中）（9） attach（接続）（10） detach（切断）（11） format（フォーマット）（12） update（更新）（13）
*ioDimmInfoWearoutIndicator	ブール値: True = デバイスの消耗しきい値を超過した
ioDimmTestMode	テスト モードをオンまたはオフにする
ioDimmTestModeIndex	選択された Fusion-io デバイスを示す番号
mib	使用中の MIB の名前
pathname	インストール時に設定されたドライバーのパス
**traps	トラップ生成をオンまたはオフにする
**updateDelay	トラップ生成のために ioMemory VSL から次の値を取得するまでの待機時間（ミリ秒）

SNMP MIB のサポート

Windows では、次の SNMP MIB フィールドがサポートされています。

fusionIoDimmMibRevMajor	fusionIoDimmInfoAdapterType
fusionIoDimmMibRevMinor	fusionIoDimmInfoAdapterPort
fusionIoDimmMIBCondition	fusionIoDimmInfoAdapterSerialNumber
fusionIoDimmInfoIndex	fusionIoDimmInfoAdapterExtPowerPresent
fusionIoDimmInfoStatus	fusionIoDimmInfoPowerlossProtectDisabled
fusionIoDimmInfoName	fusionIoDimmInfoInternalTempHigh
fusionIoDimmInfoSerialNumber	fusionIoDimmInfoAmbientTemp
fusionIoDimmInfoPartNumber	fusionIoDimmInfoPCIBandwidthCompatibility
fusionIoDimmInfoSubVendorPartNumber	fusionIoDimmInfoPCIPowerCompatibility
fusionIoDimmInfoSparePartNumber	fusionIoDimmInfoActualGoverningLevel
fusionIoDimmInfoAssemblyNumber	fusionIoDimmInfoLifespanGoverningLevel
fusionIoDimmInfoFirmwareVersion	fusionIoDimmInfoPowerGoverningLevel
fusionIoDimmInfoDriverVersion	fusionIoDimmInfoThermalGoverningLevel
fusionIoDimmInfoUID	fusionIoDimmInfoLifespanGoverningEnabled

fusionIoDimmInfoState	fusionIoDimmInfoLifespanGoverningTgtDate
fusionIoDimmInfoClientDeviceName	fusionIoDimmExtnIndex
fusionIoDimmInfoBeacon	fusionIoDimmExtnTotalPhysCapacityU
fusionIoDimmInfoPCIAddress	fusionIoDimmExtnTotalPhysCapacityL
fusionIoDimmInfoPCIDeviceID	fusionIoDimmExtnTotalLogCapacityU
fusionIoDimmInfoPCISubdeviceID	fusionIoDimmExtnTotalLogCapacityL
fusionIoDimmInfoPCIVendorID	fusionIoDimmExtnBytesReadU
fusionIoDimmInfoPCISubvendorID	fusionIoDimmExtnBytesReadL
fusionIoDimmInfoPCISlot	fusionIoDimmExtnBytesWrittenU
fusionIoDimmInfoWearoutIndicator	fusionIoDimmExtnBytesWrittenL
fusionIoDimmInfoFlashbackIndicator	fusionIoDimmExtnFormattedBlockSize
fusionIoDimmInfoWritableIndicator	fusionIoDimmExtnCurrentRAMUsageU
fusionIoDimmInfoInternalTemp	fusionIoDimmExtnCurrentRAMUsageL
fusionIoDimmInfoHealthPercentage	fusionIoDimmExtnPeakRAMUsageU
fusionIoDimmInfoMinimalModeReason	fusionIoDimmExtnPeakRAMUsageL
fusionIoDimmInfoReducedWriteReason	fusionIoDimmWearoutTrap
fusionIoDimmInfoMilliVolts	fusionIoDimmNonWritableTrap
fusionIoDimmInfoMilliVoltsPeak	fusionIoDimmFlashbackTrap
fusionIoDimmInfoMilliVoltsMin	fusionIoDimmTempHighTrap
fusionIoDimmInfoMilliWatts	fusionIoDimmTempOkTrap
fusionIoDimmInfoMilliWattsPeak	fusionIoDimmErrorTrap
fusionIoDimmInfoMilliAmps	fusionIoDimmPowerlossProtectTrap
fusionIoDimmInfoMilliAmpsPeak	

付録 H – SMI-S インターフェイス

ioMemory VSL 3.0.0 以降、SMI-S プロバイダーには `root/fio` という新しい CIM 名前空間が指定されています。

SMI-S インターフェイスの概要

SMI-S インターフェイスは、既存の Distributed Management Task Force（DMTF）および Storage Networking Industry Association（SNIA）の Storage Management Initiative Specification（SMI-S）標準に準拠しています。また、Web-Based Enterprise Management（WBEM）に基づいており、ioMemory デバイスおよび関連するソフトウェアを表す Common Information Model（CIM）モデルを提供します。このモデルでは、下位互換のある拡張が許可されており、Fusion-io によって開発される新規ハードウェアおよびソフトウェア機能に対応できます。

ここでは、WBEM、SMI-S、および DMTF 標準に精通しているユーザーを対象として説明します。このドキュメントおよび関連するモデルは、フィードバックを受けて任意の時点で変更される可能性があります。

リファレンス

CIM スキーマ v2.22
http://www.dmtf.org/standards/cim/cim_schema_v2220

DMTF DSP1011、物理的資産プロファイル
http://www.dmtf.org/standards/published_documents/DSP1011_1.0.2.pdf

DMTF DSP1023、ソフトウェア インベントリ プロファイル
http://www.dmtf.org/standards/published_documents/DSP1023_1.0.1.pdf

DMTF DSP1033、プロファイル登録プロファイル
http://www.dmtf.org/standards/published_documents/DSP1033_1.0.0.pdf

DMTF DSP1075、PCI デバイス プロファイル
http://www.dmtf.org/standards/published_documents/DSP1075_1.0.0.pdf

DMTF DSP1002、診断プロファイル
http://www.dmtf.org/standards/published_documents/DSP1002_2.0.0.pdf

SMI-S v1.4 アーキテクチャ
http://www.snia.org/sites/default/files/SMI-Sv1_4r6_Architecture_book.pdf

SMI-S v1.4 共通プロファイル
http://www.snia.org/sites/default/files/SMI-Sv1.4r6_CommonProfiles.book_.pdf

SMI-S v1.4 ホスト プロファイル
http://www.snia.org/sites/default/files/SMI-Sv1.4r6_Host.book_.pdf

SMI-S v1.4 共同診断モデル
http://www.dmtf.org/standards/mgmt/cdm/

Windows への SMI-S WMI プロバイダーのインストール

Windows に Fusion-io SMI-S WMI プロバイダーをインストールするには、次の手順に従います。

1. コントロール パネルで[プログラムの追加と削除]をクリックします。
2. [管理とモニタ ツール]をクリックして、[詳細]をクリックします。[WMI Windows インストーラ プロバイダ]が選択されていることを確認します。

ioMemory デバイスの SMI-S WMI プロバイダーがインストールされ、WMI サービスが自動的に再起動されます。

予期される警告メッセージ

WMI プロバイダーをインストールすると、Windows イベント ログに次の説明を持つ警告が表示されます。

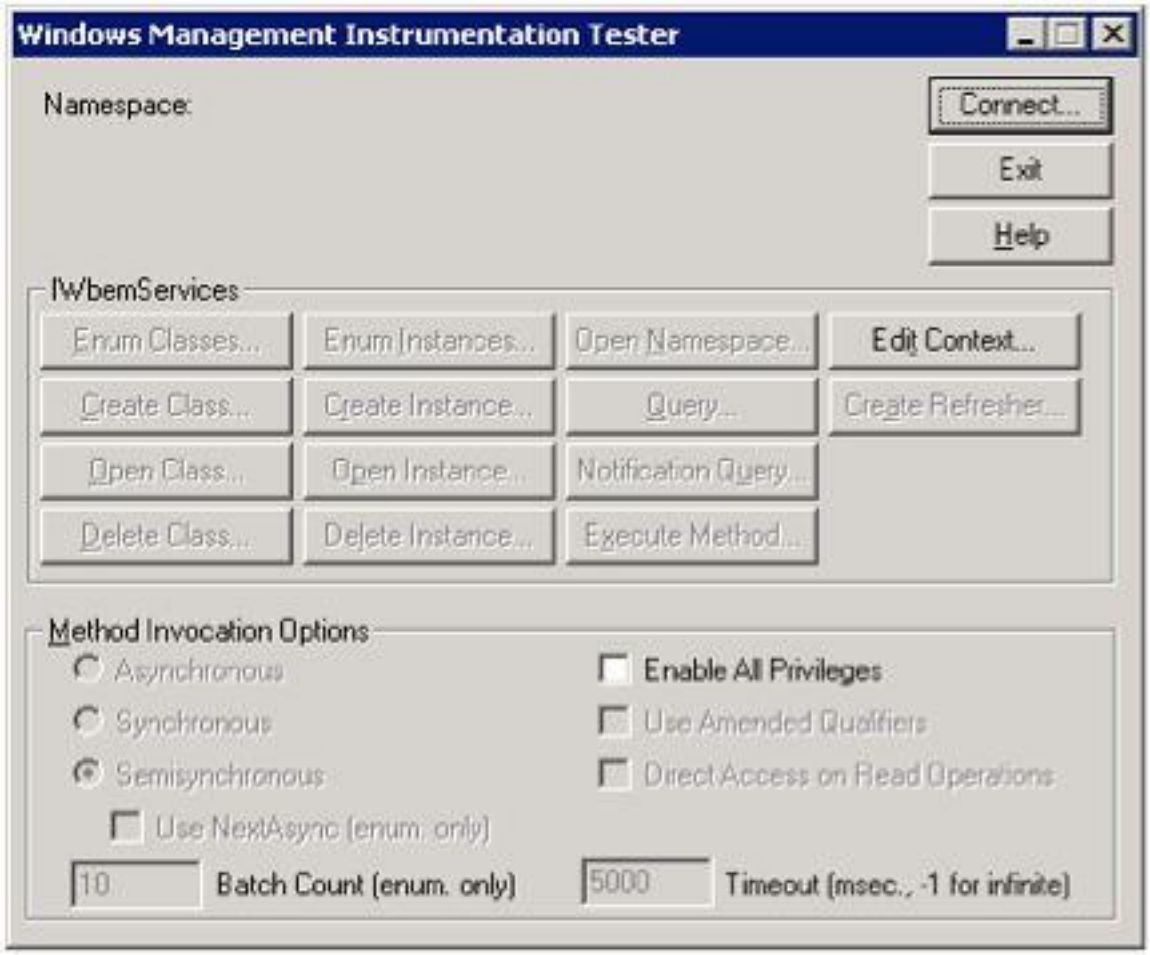
プロバイダ `fio-smis-wmi` は LocalSystem アカウントを使うために Windows Management Instrumentation 名前空間 `root\fio` に登録されました。このアカウントには特権があり、プロバイダがユーザー要求を正しく偽装しない場合はセキュリティ違反が起こる可能性があります。

この警告は、予期されたものです。WMI プロバイダーは ioMemory VSL とのみやり取りを行い、システム データは変更しません。

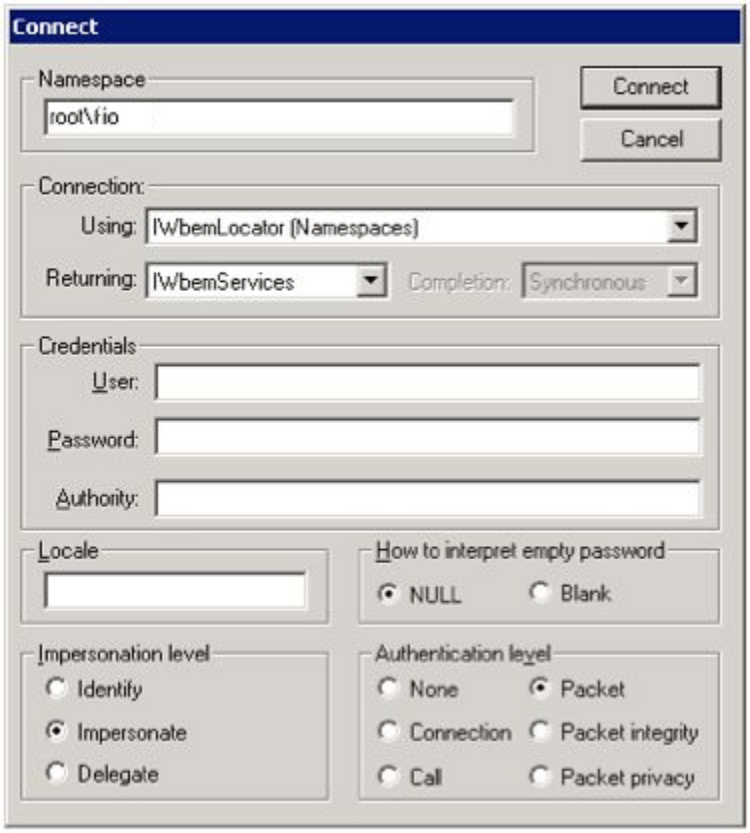
Windows における SMI-S インストールの確認

Windows で Fusion-io SMI-S WMI プロバイダーを確認するには、次の手順に従います。

1. wbemtest.exe プログラムを実行します。[Windows Management Instrumentation テスト]ウィンドウが表示されます。

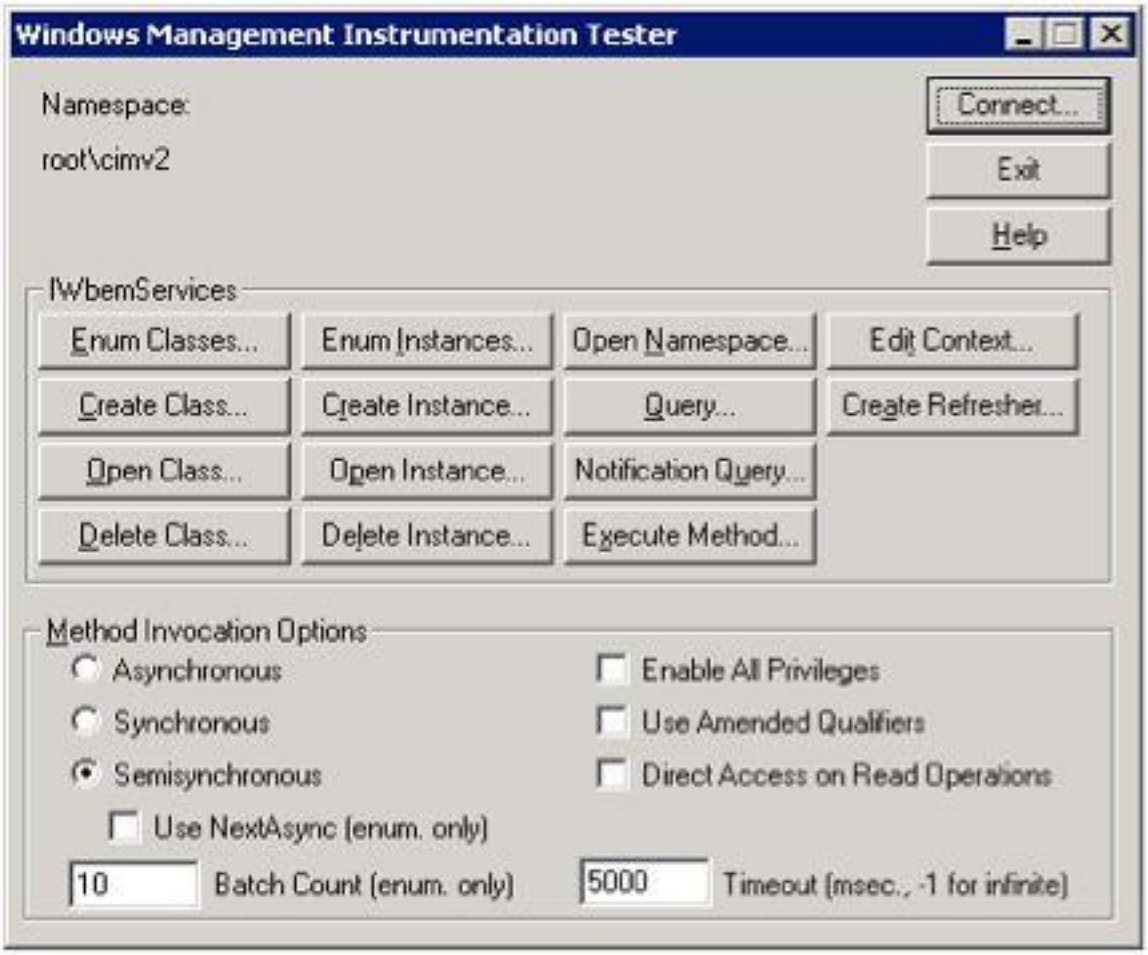


2. [接続]をクリックして、[接続]ダイアログ ボックスを表示します。CIM プロバイダーの名前空間は `root\fio` です。

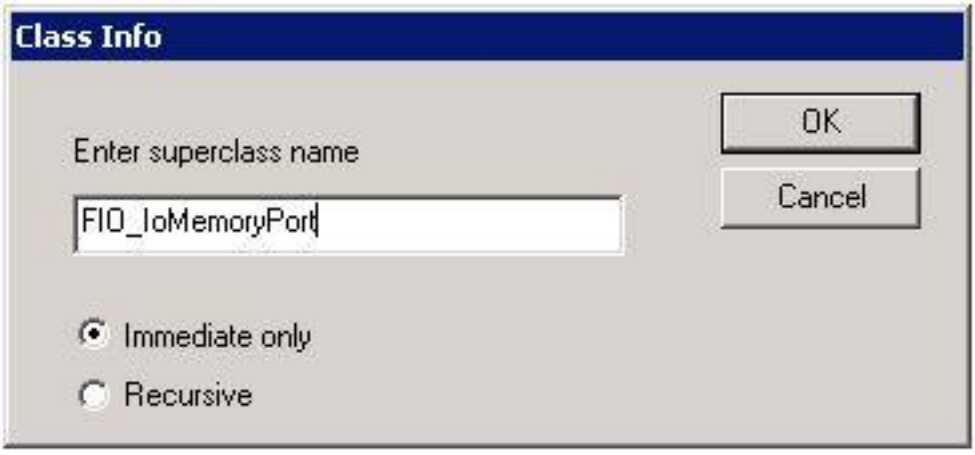


3. 上のスクリーンショットに示されている名前空間の値を入力して、[接続]をクリックします。

[Windows Management Instrumentation テスト]ウィンドウに名前空間の値が表示されます。



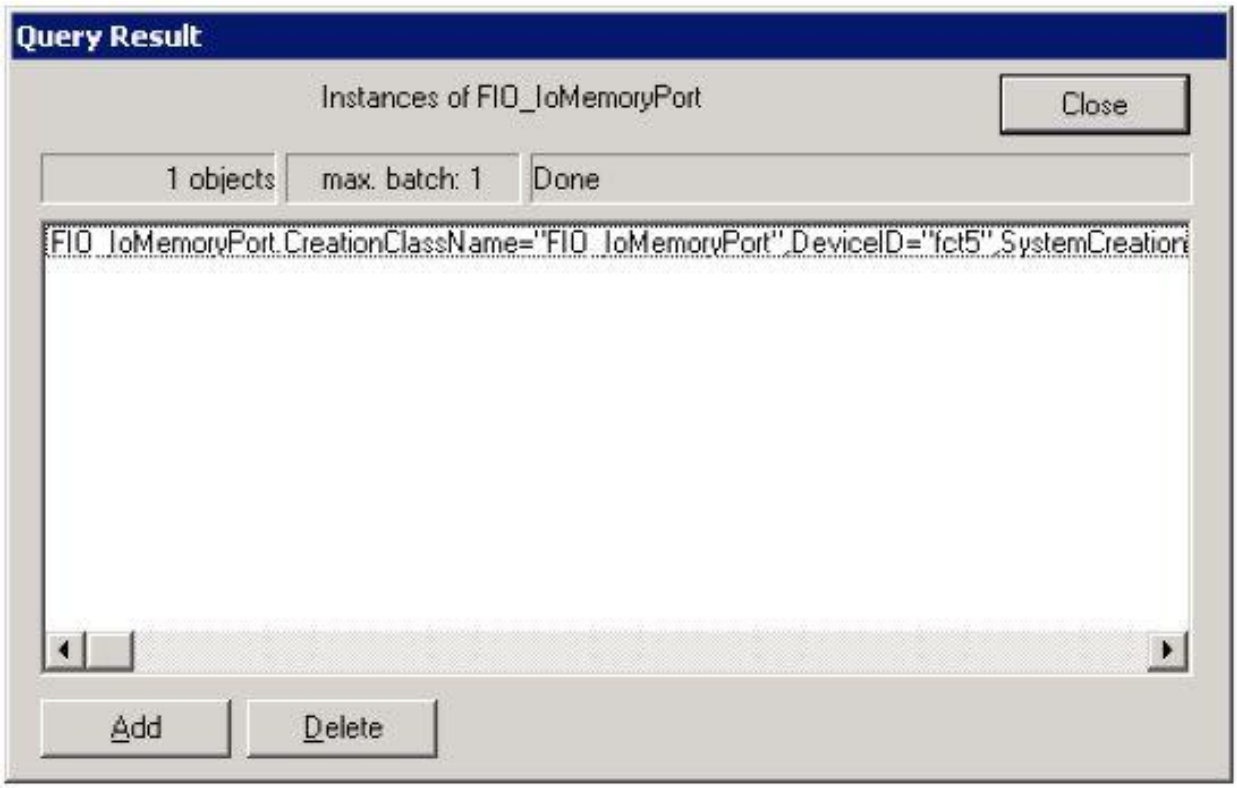
4. [インスタンスの列挙]（1 行目の 2 つ目のボタン）をクリックして、[クラス情報]ダイアログ ボックスを表示します。



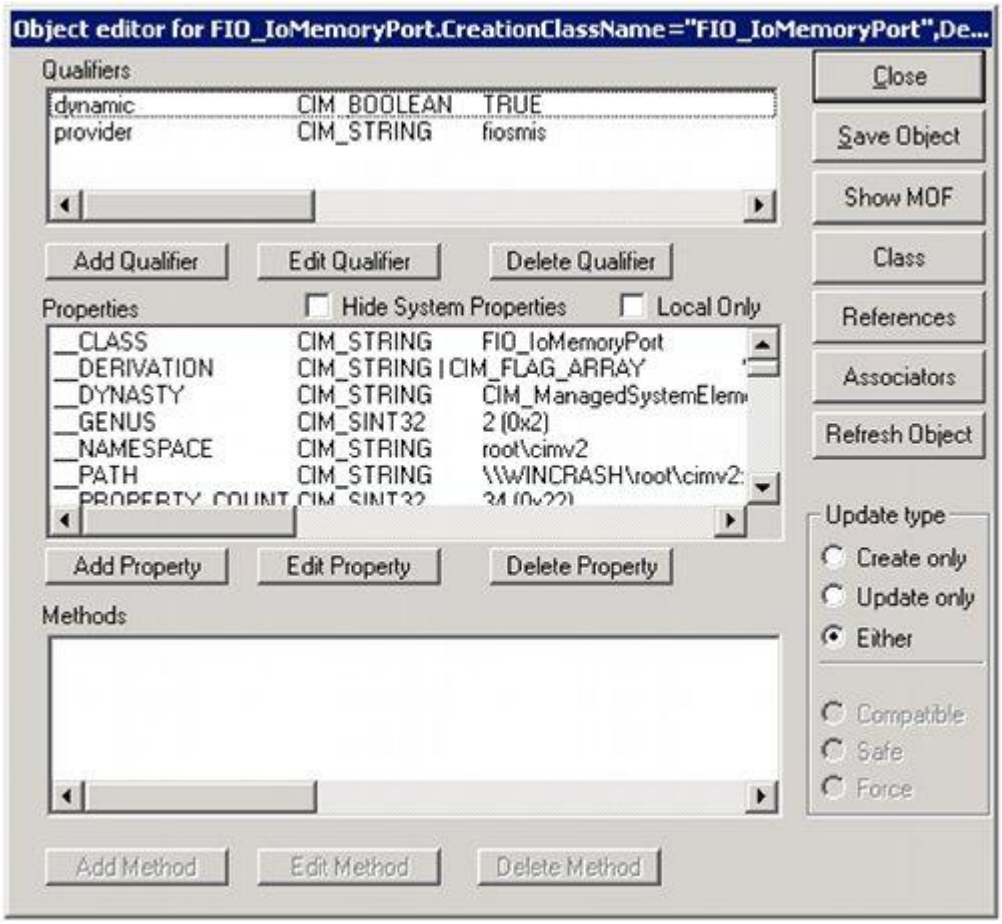
5. 上図のように「FIO_IoMemoryPort」と入力して、[OK]をクリックします。

プロバイダーが正しくインストールされていると、次の例のように結果が表示されます。システム

内にある各 ioMemory デバイスに対してエントリが表示されます。



6. エントリをダブルクリックすると、次の例のように詳細情報が表示されます。



手動登録

自動インストールによってプロバイダーを登録できない場合は、次の手順に従って手動で登録します。

1. 管理ツールの[サービス]を使用するか、または次のコマンド ラインを実行して、WMI サービス（`winmgmt`）を停止します。

   ```
   net stop winmgmt
   ```

2. コマンドライン インターフェイスを使用して `Fusion-io\SMIS\cim-schema` ディレクトリに移動し、次のコマンドを実行します。

   ```
   mofcomp fio-reg-wmi.mof
   ```

3. `Fusion-io\SMIS\WMI` ディレクトリに移動します。
4. 次のコマンドを使用して、`fio-smis-wmi.dll` を登録解除し、再度登録します。

   ```
   regsvr32 /u fio-smis-wmi.dll
   regsvr32 fio-smis-wmi.dll
   ```

5. 管理ツールの[サービス]を使用するか、または次のコマンド ラインを実行して、`winmgmt` サービスを開始します。

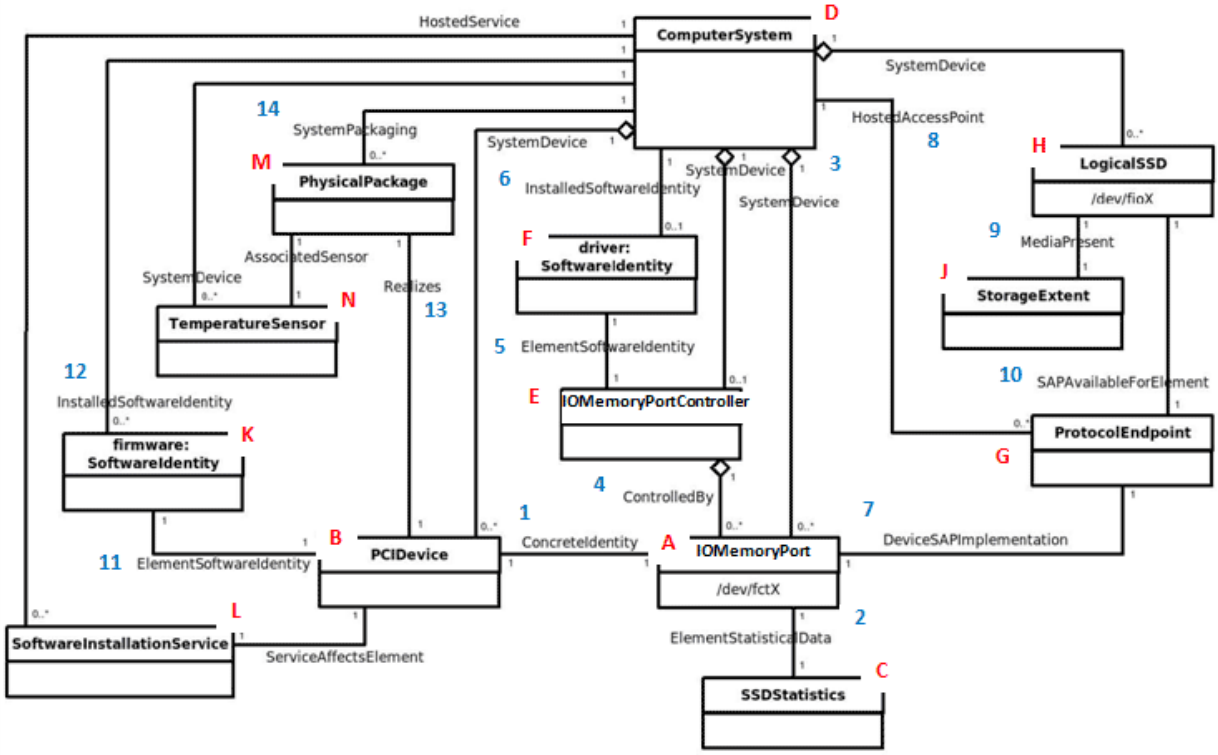
```
net start winmgmt
```

解説

SMI-S は、もともとファイバー チャネル、iSCSI、SAS などの SCSI コマンド セットに基づくストレージ エリア ネットワーク(SAN)システムに重点を置いた仕様のコレクションです。しかし、これらのストレージ システムのモデル化に使用される一般的なパターンは、Fusion-io が提供するシステムのような直接接続型のソリッドステート ストレージ システムにも適用できます。

ioMemory デバイスは、ストレージ HBA(Storage HBA)、直接接続ポート(Direct Attached(DA) Ports)、ホスト検出リソース(Host Discovered Resources) プロファイルで確立された SMI-S パターンを使用してモデル化されています。物理的資産プロファイル、ソフトウェア インベントリ プロファイル、PCI デバイス プロファイル、共同診断モデル プロファイルを含む公開済みの DMTF 仕様を使用して、ioMemory デバイスの物理的な部分、およびすべてのファームウェアと ioMemory VSL ソフトウェアがモデル化されています。

次の図に、ioMemory デバイスおよび関連するファームウェアとソフトウェアについての Fusion-io SMI-S CIM モデルを示します。見やすくするために、クラス名のプレフィクス「FIO_」は省略しています。



A: IOMemoryPort クラス

モデルの中心的なインスタンスは、ioMemory デバイスの論理的な表現である IOMemoryPort クラス（図の A）です。このクラスは、ドライブをプロビジョニングするために必要な外部メソッドをサポートしています。取り付けられた各 ioMemory デバイスには、PCIDevice（B）および IOMemoryPort のインスタンスが 1 つずつあり、これらは ConcreteIdentity（1）のインスタンスによって関連付けられています。デバイスの性能および容量に関する重要なデータを含む SSDStatistics（C）のインスタンスは、ElementStatisticalData 関連（2）によって各 IOMemoryPort に関連付けられています。IOMemoryPort は、ComputerSystem クラスのインスタンスに包含されています。SystemDevice（3）集約によって、IOMemoryPort が包含先の ComputerSystem 内に集約されています。

E: IOMemoryPortController クラス

IOMemoryPortController（E）のインスタンスは、取り付けられた ioMemory デバイスを制御するために使用される ioMemory VSL を表しています。IOMemoryPortController は CIM_PortController を特化したものであり、ControlledBy（4）集約によって IoMemoryPort を集約しています。ソフトウェア バージョンおよびベンダー情報は、ElementSoftwareIdentity（5）によって IOMemoryPortController（E）に関連付けられた SoftwareIdentity（F）のインスタンスによって表されます。インストールされている ioMemory VSL ソフトウェアを表す SoftwareIdentity は、InstalledSoftwareIdentity 関連（6）を使用して包含先の ComputerSystem に関連付けられています。

ProtocolEndpoint クラス（G）のインスタンスは、IOMemoryPort とソリッドステート ストレージとの間の論理データ パスの両端を表しています。モデルのこの部分は、ポートがイニシエーターおよびターゲットの両方の役割を持つ DA ポート プロファイルのパターンから導出されています。ProtocolEndpoint は、DeviceSAPImplementation（7）によって IOMemoryPort に、HostedAccessPoint（8）によって ComputerSystem に関連付けられています。

H: LogicalSSD クラス（ブロック デバイス）

アプリケーション（ファイル システム、データベース、および論理ボリューム マネージャー）に公開されるブロック デバイスは、CIM_DiskDrive のサブクラスである LogicalSSD（H）のインスタンスを使用してモデル化されています。このクラスは MediaPresent 関連（9）を使用して StorageExtent（J）に関連付けられていますが、StorageExtent は常に存在しています。また、SAPAvailableForElement（10）を使用して IOMemoryPort を表す ProtocolEndpoint（G）に、SystemDevice（3）を使用して包含先の ComputerSystem に関連付けられています。

PCIe デバイスである ioMemory デバイスは、PCIDevice クラス（B）のインスタンスによっても表されます。IOMemoryPort は PCIDevice および関連する制御デバイスの代替表現であり、これらは ConcreteIdentity 関連によって関連付けられています。

K: SoftwareIdentity

ioMemory VSL ソフトウェアは、SoftwareIdentity によっても表されます。このクラスは、ElementSoftwareIdentity 関連(11)によって PCIDevice に関連付けられています。SoftwareIdentity(ファームウェア)は、InstalledSoftwareIdentity 関連(12)によって包含先の ComputerSystem に関連付けられています。各 PCIDevice には、デバイスのファームウェアの更新に使用できる SoftwareInstallationService(L)のインスタンスが関連付けられています。

M: 物理的な部分

ioMemory デバイスの物理的な部分は、PhysicalPackage クラス(M)のインスタンスによって表されます。このクラスは、Realizes(13)によって PCIDevice に、SystemPackaging(14)によって包含先の ComputerSystem に関連付けられています。ioMemory デバイスの温度センサーは、TemperatureSensor(N)のインスタンスによって表され、AssociatedSensor によって PhysicalPackage に関連付けられています。

実装

ここでは、Fusion-io デバイスの CIM モデルの各種インスタンスおよび関連の規約について説明します。ここでは、クラスのプロパティすべてについては詳細に説明しません。すべてのプロパティの詳細説明については、CIM スキーマを参照してください。

このモデルに基づく WBEM CIM プロバイダーは、将来開発予定です。Fusion-io では、OpenPegasus、OpenWBEM、SFCB、Windows WMI などのよく使われる CIMOM をサポートする予定です。

デバイスの正常性は、HealthLevel プロパティの値によって示されます。値には、正常、警告、書き込み低減、読み取り専用などがあります。これらの値は、適宜 standardHealthState の値(OK、低下/警告、および重大な障害)にマッピングされます。

デバイスのプロビジョニング用外部メソッドには、attach(接続)、detach(切断)、format(フォーマット)、および update(更新)があります。attach(接続)メソッドでは、ioMemory デバイスのブロック デバイスが作成されます。detach(切断)では、ブロック デバイスが無効化されます。format(フォーマット)メソッドを使用すると、メガバイト単位またはパーセンテージでデバイスのサイズを指定できます。update(更新)メソッドを使用すると、デバイス上のファームウェアをアップグレードできます。

デバイスの寿命は、HealthPercentage プロパティの値によって示されます。FlashbackAvailability は、ioMemory デバイスの Flashback 機能がオンラインであるかどうかを示します。

IOMemoryPort は、ControlledBy 集約によって IOMemoryPortController 内に集約されています。IOMemoryPort のインスタンスは、ConcreteIdentity 関連によって対応する PCIDevice に関

連付けられています。IOMemoryPort は、包含先の ComputerSystem の論理デバイスであり、SystemDevice 集約によってそのことが示されています。

ioDrive Duo デバイスなど、複数の ioMemory デバイスで構成された製品は、2 つの異なる ioMemory デバイスであるかのように表現されます。複数のデバイスで構成された製品では、IOMemoryPort クラスが拡張されて、装着されているカードのタイプ、シリアル番号、および外部電源接続についての製品全体としての情報が格納されます。

IOMemoryPort

ComputerSystem に取り付けられている各 ioMemory デバイスに対して、IOMemoryPort のインスタンスが 1 つ存在します。

LocationIndicator プロパティは、デバイスのインジケーター ビーコンの状態（すべての LED が点灯しているなど）を反映しています。この値を読み取ることで、インジケーターの現在の状態を把握できます。「オン」または「オフ」の値を書き込むと、インジケーターがオンまたはオフになるため、デバイスの物理的な場所を特定するために使用できます。

SSDStatistics

IOMemoryPort の各インスタンスに対して、SSDStatistics のインスタンスが 1 つ存在します。このオブジェクトのプロパティによって、性能および容量に関するデータが提供されます。一部の情報は、デバイスが接続されているとき（つまり、関連する IOMemoryPort の状態が「接続済み」のとき）にのみ取得できます。

IOMemoryPortController

IOMemoryPortContoller は、IOMemoryPort を制御するために使用される ioMemory VSL を表しており、1 つのインスタンスのみ存在しています。IOMemoryPortController は、CIM_PortController を特化したものです。

IOMemoryPortController は、SystemDevice 集約を使用して、包含先の ComputerSystem に集約されています。IOMemoryPortController は、ElementSoftwareIdentity 関連によって、ioMemory VSL ソフトウェアのプロパティを表す SoftwareInventory のインスタンスに関連付けられています。

ProtocolEndpoint

IOMemoryPort の各インスタンスに対して、ProtocolEndpoint のインスタンスが 1 つ存在します。DeviceSAPImplementation を使用して IOMemoryPort に、SAPAvailableForElement を使用して LogicalSSD に関連付けられています。IOMemoryPort は、イニシエーター ポートとターゲット ポート

の両方を表しているため、IOMemoryPort と LogicalSSD との間の接続をモデル化する場合は、1 つの IOMemoryPort につき 1 つの ProtocolEndpoint のみが必要となります。

LogicalSSD

ioMemory デバイスによって公開される各ブロック デバイス(/dev/fioX)に対して、CIM_DiskDrive のサブクラスである LogicalSSD のインスタンスが 1 つ存在します。オペレーティング システムのデバイス名に基づいて、関連性のある ID が使用されます。これにより、クライアント アプリケーションは、このモデルで検出したブロック デバイスを、ホスト システム上に備えられた他の SMI-S モデルから検出されたリソースに関連付けることができます。

ComputerSystem は、SystemDevice によって LogicalSSD を集約しています。LogicalSSD のインスタンスは、SAPAvailableForElement によって対応する ProtocolEndpoint に関連付けられています。エンドポイントに関連付けられている IOMemoryPort が接続されていない場合は、Availability プロパティが「オフライン」に設定され、DeviceID プロパティの値が「不明」に設定されます。

StorageExtent

StorageExtent の 1 つのインスタンスが各 LogicalSSD に関連付けられており、関連するデバイスの論理ストレージを表しています。

SoftwareIdentity

ioMemory VSL ソフトウェアを表す SoftwareIdentity のインスタンスが 1 つ存在しています。ファームウェアも SoftwareIdentity を使用してモデル化されますが、取り付けられている各 ioDrive に対して 1 つのインスタンスが必要です。IsEntity プロパティの値は True であり、SoftwareIdentity のインスタンスが、ioMemory VSL ソフトウェアまたはファームウェアの個別のコピーに対応していることを示しています。MajorVersion、MinorVersion、RevisionNumber、および BuildNumber プロパティは、ドライバーおよびファームウェアのバージョン情報を含んでいます。Manufacturer プロパティは、Fusion-io を識別するために使用できます。

ファームウェアは実際には ComputerSystem にインストールされるわけではないため、ComputerSystem との InstalledSoftwareIdentity 関連を省略してファームウェアをモデル化することもできます。この方法を使用するかどうかは、ファームウェアをどのようにモデル化する必要があるかによって異なります。

SoftwareInstallationService

各 PCIDevice に対して SoftwareInstallationService のインスタンスが 1 つ存在し、関連するデバイスのファームウェアの更新に使用できます。

PCIDevice

コンピューター内の各 ioMemory デバイス(PCIe カード)に対して 1 つの PCIDevice がインスタンス化されます。プロパティは次のように設定されます。

- BusNumber – PCIe デバイスが存在するバス番号です。
- DeviceNumber – このバスの PCI デバイスに割り当てられたデバイス番号です。
- FunctionNumber – PCI デバイスの機能番号に設定されます。
- SubsystemID、SubsystemVendorID、PCIDeviceID、VendorID、および RevisionID はオプションですが、PCI デバイスの構成レジスターから値を抽出できる場合は、値を指定できます。

PCIDevice は、ConcreteIdentity を使用して代替論理表現である IOMemoryPort と関連付けられています。また、PCIDevice は、Realizes によって、ioDrive の物理的な部分を表す PhysicalPackage とも関連付けられています。

PhysicalPackage

コンピューター システムに取り付けられている個別の各物理 ioDrive に対して、PhysicalPackage のインスタンスが 1 つ存在します。Manufacturer、Model、SKU、SerialNumber、Version、および PartNumber プロパティを使用して、物理カードの対応する属性を記述できます。PhysicalPackage は、Realizes によって PCIDevice に、SystemPackaging によって包含先の ComputerSystem に関連付けられています。

TemperatureSensor

各 PhysicalPackage に対して TemperatureSensor のインスタンスが 1 つ存在します。ドライブの温度情報は、このオブジェクトのプロパティに保存されます。

DiagnosticTest

DiagnosticTest の 1 つのインスタンスが存在しています。RunDiagnostic() メソッドによって、指定された ManagedElement(IoMemoryPort のインスタンスである必要があります)のデバイス ステータスのスナップショットの取得が開始されます。診断の実行は同期的に行われ、即時に実行されます。結果として得られる ConcreteJob オブジェクトは、元の DiagnosticTest インスタンス、および指定された各 IoMemoryPort インスタンスに関連付けられます(図 2 を参照)。現時点では、RunDiagnostic()は、デフォルトの DiagnosticSettingData を指定した場合にのみ使用できます。

実行するごとに、DiagnosticSettingDataRecord および関連する DiagnosticCompletionRecord のエントリが DiagnosticLog に 1 つ追加されます。DiagnosticCompletionRecord の RecordData プロパティには、実行時の重要なデバイス ステータスが記録されます。RecordData 文字列の形式は、RecordFormat プロパティに指定されています。

この文字列は、「Unknown（不明）」、「OK」、「Warning（警告）」、または「Error（エラー）」のいずれかの値を含む一連のステータス文字列がアスタリスク（*）文字で区切られたフォーマットになっています。現在、WearoutStatus、WritabilityStatus、FlashbackStatus、TemperatureStatus、MinimalModeStatus、PciStatus、および InternalErrorStatus の 7 つのステータス値が記録されます。通常の動作状態においては、これらすべてが「OK」と報告されます。

デバイスに確保された残りの領域が 10%未満になると、WearoutStatus が「Warning（警告）」に設定されます。確保された領域がなくなると、「Error（エラー）」に設定されます。

- デバイスでスロットリング（write trhottling）による書き込み調整が行われている場合、またはデバイスが読み取り専用モード（read-only mode）の場合は、WritabilityStatus が「Error（エラー）」に設定されます。デバイスの消耗や電力不足など、さまざまな条件によってこの状態になる場合があります。
- 回復不可能なエラーによって Flashback Protection 機能が低下している場合は、FlashbackStatus に「Warning（警告）」が報告されます。
- デバイスの温度が最大安全温度に近づいている場合は、TemperatureStatus に「Warning（警告）」が報告されます。最大安全温度に達したか、または最大安全温度を超えた場合は、「Error（エラー）」が報告されます。
- デバイスが最小モード（minimal mode）の場合は、MinimalModeStatus に「Warning（警告）」または「Error（エラー）」が報告されます。
- ホストの PCIe バスに互換性の問題がある場合は、PciStatus に「Warning（警告）」または「Error（エラー）」が報告されます。
- ioMemory VSL に内部的な問題が発生している場合は、InternalErrorStatus に「Error（エラー）」が報告されます。

CompletionState プロパティには、結果の要約が設定され、「Unknown（不明）」、「OK」、「Warning（警告）」、または「Failed（エラー）」が設定されます。いずれかのステータスがエラーの場合、結果の状態には「Failed（エラー）」が報告されます。それ以外の場合で、いずれかのステータスが警告のときは、結果の状態に「Warning（警告）」が報告されます。Message プロパティには、警告やエラーがある場合の適切な対処方法を示す文字列が設定されます。

DiagnosticSettingData

DiagnosticTest のインスタンスに関連付けられた DiagnosticSettingData のインスタンスが 1 つあります。このインスタンスには、RunDiagnostic の各呼び出しに対するデフォルト設定が記録されています。

DiagnosticServiceCapabilities

DiagnosticTest のインスタンスに関連付けられた DiagnosticServiceCapabilities のインスタンスがあり、DiagnosticTest サービスの機能が記録されます。

DiagnosticLog

DiagnosticLog のインスタンスは DiagnosticTest のインスタンスに関連付けられており、各実行の結果が保存されます。

DiagnosticSettingRecord

診断が実行されるたびにデフォルトの DiagnosticSettingData のコピーが DiagnosticSettingDataRecord に保存され、DiagnosticCompletionRecord のインスタンスと関連付けられます。

DiagnosticCompletionRecord

DiagnosticCompletionRecord のインスタンスには、RunDiagnostic の各実行の結果が保存されます。詳細については、DiagnosticTest の説明を参照してください。

RegisteredDiskDriveLiteProfile

このクラスのインスタンスは、1 つのみ必要です。/root/interop 名前空間にあり、ディスク ドライブ ライト プロファイル（Disk Drive Lite Profile）の実装を示しています。プロパティは次のように設定されます。

- InstanceID – 「SNIA:DiskDriveLiteProfile-1.4.0」に設定されます。
- RegisteredOrganization – 「11」（SNIA）に設定されます。
- RegisteredName – 「DirectAccess Ports Profile」に設定されます。
- RegisteredVersion – 「1.4.0」に設定されます。

RegisteredDAPortsProfile

このクラスのインスタンスは、1 つのみ必要です。/root/interop 名前空間にあり、DA ポート プロファイルの実装を示しています。プロパティは次のように設定されます。

- InstanceID – 「SNIA:DAPortsProfile-1.4.0」に設定されます。
- RegisteredOrganization – 「11」（SNIA）に設定されます。
- RegisteredName – 「DirectAccess Ports Profile」に設定されます。
- RegisteredVersion – 「1.4.0」に設定されます。

RegisteredStorageHBAProfile

このクラスのインスタンスは、1 つのみ必要です。/root/interop 名前空間にあり、ストレージ HBA（Storage HBA）プロファイルの実装を示しています。プロパティは次のように設定されます。

- InstanceID – 「SNIA:StorageHBAProfile-1.4.0」に設定されます。
- RegisteredOrganization – 「11」（SNIA）に設定されます。

- RegisteredName – 「Storage HBA Profile」に設定されます。
- RegisteredVersion – 「1.4.0」に設定されます。

RegisteredHostDiscoveredResourcesProfile

このクラスのインスタンスは、1 つのみ必要です。/root/interop 名前空間にあり、ホスト検出リソース（Host Discovered Resources）プロファイルの実装を示しています。プロパティは次のように設定されます。

- InstanceID – 「SNIA:HostDiscoveredResourcesProfile-1.2.0」に設定されます。
- RegisteredOrganization – 「11」(SNIA)に設定されます。
- RegisteredName – 「Host Discovered Resources Profile」に設定されます。
- RegisteredVersion – 「1.2.0」に設定されます。

RegisteredPCIDeviceProfile

このクラスのインスタンスは、1 つのみ必要です。/root/interop 名前空間にあり、PCI デバイス プロファイルの実装を示しています。プロパティは次のように設定されます。

- InstanceID – 「DMTF:DSP1075-PCIDevice-1.0.0a」に設定されます。
- RegisteredOrganization – 「2」(DMTF)に設定されます。
- RegisteredName – 「PCIDevice Profile」に設定されます。
- RegisteredVersion – 「1.0.0a」に設定されます。

RegisteredSoftwareInventoryProfile

このクラスのインスタンスは、1 つのみ必要です。/root/interop 名前空間にあり、ソフトウェア インベントリ プロファイルの実装を示しています。プロパティは次のように設定されます。

- InstanceID – 「DMTF:DSP1023-SoftwareInventory-1.0.1」に設定されます。
- RegisteredOrganization – 「2」(DMTF)に設定されます。
- RegisteredName – 「Software Inventory Profile」に設定されます。
- RegisteredVersion – 「1.0.1」に設定されます。

RegisteredSoftwareUpdateProfile

このクラスのインスタンスは、1 つのみ必要です。/root/interop 名前空間にあり、ソフトウェア更新プロファイル（Software Update Profile）の実装を示しています。プロパティは次のように設定されます。

- InstanceID – 「DMTF:DSP1023-SoftwareUpdate-1.0.0」に設定されます。
- RegisteredOrganization – 「2」(DMTF)に設定されます。
- RegisteredName – 「Software Update Profile」に設定されます。

- RegisteredVersion – 「1.0.0」に設定されます。

RegisteredPhysicalAssetProfile

このクラスのインスタンスは、1 つのみ必要です。/root/interop 名前空間にあり、物理的資産プロファイルの実装を示しています。プロパティは次のように設定されます。

- InstanceID – 「DMTF:PhysicalAssetProfile-1.0.2」に設定されます。
- RegisteredOrganization – 「2」(DMTF)に設定されます。
- RegisteredName – 「PhysicalAsset Profile」に設定されます。
- RegisteredVersion – 「1.0.2」に設定されます。

RegisteredSensorsProfile

このクラスのインスタンスは、1 つのみ必要です。/root/interop 名前空間にあり、センサー プロファイル(Sensors Profile)の実装を示しています。プロパティは次のように設定されます。

- InstanceID – 「SNIA:SensorsProfile-1.0.0」に設定されます。
- RegisteredOrganization – 「11」(SNIA)に設定されます。
- RegisteredName – 「Sensors Profile」に設定されます。
- RegisteredVersion – 「1.0.0」に設定されます。

RegisteredCommonDiagnosticProfile

このクラスのインスタンスは、1 つのみ必要です。/root/interop 名前空間にあり、共同診断モデル プロファイルの実装を示しています。InstanceID プロパティの値は、「DMTF:DiagnosticsProfile-2.0.0a」に設定されます。RegisteredOrganization プロパティの値は、「2」(DMTF)に設定されます。RegisteredName プロパティの値は、「Diagnostics Profile」に設定されます。RegisteredVersion プロパティの値は、「2.0.0a」に設定されます。

通知

特定の ioMemory デバイスに重大な状態が発生している場合は、定期的に通知が生成されます。現在、WBEM プロバイダーでは、6 種類の通知がサポートされています。これらによって、SMI-S プロバイダーのユーザーに対して、間もなく発生する消耗、書き込み性能の低下、Flashback 機能の低下、温度の上昇、内部エラー状態などの状態について通知されます。

通知は、CIM_AlertIndication クラスを特化した FIO_AlertIndication クラスのインスタンスです。

FIO_AlertIndication のインスタンスのプロパティの値は現在策定中であり、テストの進行およびフィードバックの受領にともなって変更される可能性があります。

FIO_AlertIndication

プロパティ	値
IndicationIdentifier	それぞれのタイプについて後の説明を参照
IndicationTime	送信された時点のタイムスタンプ
AlertingManagedElement	IoMemoryPort.DeviceID=<デバイス ID>
AlertingElementFormat	CIMObjectPath(2)
AlertType	デバイス アラート(5)
PerceivedSeverity	それぞれのタイプについて後の説明を参照
ProbableCause	それぞれのタイプについて後の説明を参照
SystemCreationClassName	「FIO_AlertIndication」
SystemName	<ホスト名>
ProviderName	「fiosmis」
CorrelatedIndications	未使用
Description	クラスの説明
OtherAlertType	未使用
OtherSeverity	未使用
ProbableCauseDescription	未使用
EventID	IndicationIdentifier と同じ
OwningEntity	<ベンダー>
MessageID	未定
Message	未定
MessageArguments	未定

書き込み性能低減通知

ioMemory VSL では、過度の消耗、温度の上昇、電力不足などのデバイスの状態を管理するために、動的に書き込みスループットが制限される場合があります。書き込み性能低減通知は、デバイスが書き込み低減モード(write-reduced mode)である間に生成されます。このモードの原因が過度の消耗である場合、IoMemoryPort の正常性パーセンテージでは正常性が 0%と報告されます。

プロパティ	値
IndicationIdentifier	<mfr>":"<ホスト名>":write"
PerceivedSeverity	低下/警告(3)
ProbableCause	しきい値超過(52) 温度の許容範囲超過(51)

	電力の問題(36)

読み取り専用通知

ドライブが寿命に達すると、書き込むことができなくなり、読み取りのみを行うことができるようになります。このような場合に読み取り専用通知が送信されます。この場合、IoMemoryPort の正常性パーセンテージでは、引き続き正常性が 0%と報告されます。

プロパティ	値
IndicationIdentifier	<mfr>":"<ホスト名>":read_only"
PerceivedSeverity	低下/警告(3)
ProbableCause	しきい値超過(52)

消耗通知

ドライブが消耗した場合、ドライブの正常性パーセンテージが 10%未満に低下した後、書き込みスループットが制限される前に、この通知が警告として生成されます。

プロパティ	値
IndicationIdentifier	<mfr>":"<ホスト名>":wearout"
PerceivedSeverity	低下/警告(3)
ProbableCause	しきい値超過(52)

Flashback 通知

部品の回復不可能な障害によって Flashback 機能の効率性が低下した場合は、この通知が送信されます。

プロパティ	値
IndicationIdentifier	<mfr>":"<ホスト名>":flashback"
PerceivedSeverity	低下/警告(3)
ProbableCause	冗長性の消失(88)

温度上昇通知

カードの温度が過度に上昇した場合にこの通知が送信されます。

プロパティ	値
IndicationIdentifier	<mfr>":"<ホスト名>":temperature"

PerceivedSeverity	重大(6)
ProbableCause	温度の許容範囲超過(51)

エラー通知

ioMemory VSL がエラー状態になった場合は、エラー通知が送信されます。

プロパティ	値
IndicationIdentifier	<mfr>":"<ホスト名>":error"
PerceivedSeverity	重大(6)
ProbableCause	その他(1)

付録 I – NUMA 構成

NUMA アーキテクチャについて

NUMA（Non-Uniform Memory Access）アーキテクチャを備えたサーバーでは、ioMemory デバイスの性能を最大限発揮するために特別なインストール手順が必要です。このようなサーバーには、HP DL580、HP DL980、IBM 3850 サーバーなどがあります。

NUMA アーキテクチャを備えたサーバーでは、システムの起動時に BIOS によって NUMA ノード間で PCIe スロットが均等に配分されない場合があります。各 NUMA ノードには、複数の CPU が含まれています。スロットが均等に配分されないと、作業負荷が高まった場合に、半分以上の CPU がアイドル状態であるにもかかわらず、残りの CPU の使用率が 100%になる可能性があります。このような不均等を回避するために、利用可能な NUMA ノードに対して、ioMemory デバイスを手動で均等に割り当てる必要があります。

以降の例では、カスタム アフィニティ設定の最終的な実装を示しています。実装では、システムのアーキテクチャ、取り付けられている ioMemory デバイスの種類と数、使用されている特定の PCIe スロットなどの、特定のシステムについての分析が必要になります。特定の環境に対して、そのセットアップに応じたカスタムの分析が必要です。このような分析を行う場合は、特定のインストールに合わせてシステムの NMUA アーキテクチャを理解する必要があります。

サーバーの構成によって、実際の設定は次の例と異なる場合があります。実際のシステムで正しい設定を行うために、`fio-status` を使用して、すべてのデバイス（fct 番号）をリスト表示します。次に、`fio-beacon` を使用して、各 PCIe スロットのそれぞれのデバイスを識別します。その後、次の例に示す `fio-config` 構文の例をテンプレートとして使用し、実際のシステムに合わせて変更します。

NUMA アーキテクチャを採用したサーバーで ioMemory デバイスを構成するには、`FIO_AFFINTIY` パラメータを指定して `fio-config` ユーティリティを使用する必要があります。

FIO_AFFINITY パラメータ

`FIO_AFFINITY` パラメータは、システム内のすべてのアダプターのアフィニティ設定を指定する、3 つの項目（triplets）が一組になった<アフィニティの指定>のリストです。3 つの項目のそれぞれはコンマで区切り、3 つの項目の各組はセミコロンで区切ります。

構文:

```
fio-config -p FIO_AFFINITY <アフィニティ指定>[;<アフィニティ指定>...]
```

各<アフィニティ指定>の構文は次のとおりです。

<fct 番号>,[g|n]<グループまたはノード番号>[,<16 進マスク>]

グループ/ノード番号の前に g または n 文字がない場合、番号はグループ番号であるとみなされます。

16 進マスクの指定は任意です。指定しない場合、マスクは 0xffffffffffffffff であるとみなされます。0x プレフィクスの指定も任意です。

16 進マスクがノードのマスクの場合、マスクはノードが属するグループではなく、そのノードのみに対する指定となります。

簡単な例:

```
fio-config -p FIO_AFFINITY 4,n1,0xf;5,n0;7,g1;9,g2,0xff0
```

この例では、次のアフィニティが作成されます。

デバイス	ノード/グループ	プロセッサーのアフィニティ
fct4	ノード 1	ノード内のプロセッサー0～3（マスク 0xf）
fct5	ノード 0	ノード内のすべてのプロセッサー（16 進マスクなし）
fct7	グループ 1	グループ内のすべてのプロセッサー（16 進マスクなし）
fct9	グループ 2	グループ内のプロセッサー4～11（マスク 0xff0）

高度な構成の例

このサンプルのサーバーは、4 つの NUMA ノードで構成されており、ノードあたり 8 つのハーパースレッドコアがあります（ノードあたり 16 の論理プロセッサー、システム全体で 64 の論理プロセッサーがあります）。また、このシステムでは拡張構成が使用されており、11 の PCIe 拡張スロットがあります。システムの起動時に、システムの BIOS によって PCIe スロット 1～6 が NUMA ノード 2 に、PCIe スロット 7～11 が NUMA ノード 0 に割り当てられます。NUMA ノード 1 および 3 には、PCIe スロットは割り当てられません。このように割り当てた場合、ioMemory デバイスのトラフィックが非常に多くなったときに、システムで負荷分散に関する問題が発生します。具体的には、デバイスの使用率が高くなると、システム内の CPU の半分がアイドル状態であるにもかかわらず、残りの半分の CPU の使用率が 100%になり、ioMemory デバイスのスループットが制限されます。

この問題を回避するためには、FIO_AFFINITY 構成パラメータを使用して ioMemory デバイスのアフィニティを手動で構成し、すべての NUMA ノードに作業負荷を分散する必要があります。このパラメータを指定すると、ioMemory VSL ドライバーのデフォルトの動作が上書きされます。FIO_AFFINITY 構成パラメータの詳細については、上記の構文説明を参照してください。

次に、10 の ioDrive Duo デバイスを手動で構成する例を示します（各 ioDrive Duo デバイスは、2 つの ioMemory デバイスで構成されています）。スロット 1 は Gen1 PCIe スロットなので、ioDrive Duo デバイスとは互換性がありません。したがって、スロット 2～11 に ioDrive Duo デバイスを取り付けます。

各 ioDrive Duo デバイスは 2 つの ioMemory デバイスで構成されているため、それぞれの ioDrive Duo デバイスに対して 2 つのデバイス番号があります（各 ioMemory デバイスに対して 1 つの番号）。したがって、各スロットに 2 つのデバイス番号があります。

システム起動時の BIOS によるデフォルトの NUMA ノード割り当ては次のとおりです。

BIOS によって割り当てられる NUMA ノード	PCIe スロット	FCT デバイス番号	プロセッサーのアフィニティ
0	7-11	8,9,13,14,18,19,23,24,28,29	ノード内のすべてのプロセッサー
1	なし	なし	なし
2	2-6	135,136,140,141,145,146,150,151,155,156	ノード内のすべてのプロセッサー
3	なし	なし	なし

BIOS によるデフォルトの割り当てでは、カードがシステム内の 2 つの NUMA ノードにしか割り当てられないため、負荷が均等に分散されません。作業負荷を均等に分散するために、次の手動設定を行います。

割り当てられる NUMA ノード	PCIe スロット	FCT デバイス番号	プロセッサーのアフィニティ
0	7-9	8,9,13,14,18,19	ノード内のすべてのプロセッサー（16 進マスクなし）
1	10-11	23,24,28,29	ノード内のすべてのプロセッサー（16 進マスクなし）
2	2-3	135,136,140,141	ノード内のすべてのプロセッサー（16 進マスクなし）
3	4-6	145,146,150,151,155,156	ノード内のすべてのプロセッサー（16 進マスクなし）

ioMemory VSL ドライバーにこれらの上書き設定を構成するには、次のパラメータを指定して `fio-config` を実行します。

```
fio-config -p FIO_AFFINITY
8,n0;9,n0;13,n0;14,n0;18,n0;19,n0;23,n1;24,n1;28,n1;29,n1;
135,n2;136,n2;140,n2;141,n2;145,n3;146,n3;150,n3;151,n3;155,n3;156,n3
```

上記の例では、見やすくするために改行が挿入されています。FIO_AFFINITY の実際の実装では、改行は挿入しません。

この例では、どのデバイスにも<16 進マスク>が指定されていないことに注意してください（<アフィニティの指定>は、3 つの項目ではなく 2 つの項目（couplet）が一組になっています）。これは、各デバイスが、指定されたノードのすべてのプロセッサーで共有されることを意味しています。構成を微調整する場合は、各<アフィニティの指定>で 16 進マスクを指定して、各デバイスに対するプロセッサーの設定ができます。

エラーがないかログを確認する

有効ではない構成を入力した場合は、設定が無効になり、システム ログにエラーが表示されます。

例:

```
fio-config -p FIO_AFFINITY 5,g0,0xf;6,0xf
```

この例では、マスクの前にグループ/ノード番号が指定されていないため、デバイス fct6 のアフィニティが正しく設定されていません。システム ログに次のエラーが表示されます。

```
2011-09-09T12:22:15.176086800Z - ERROR - FusionEventDriver - FIO_AFFINITY:
Invalid group or node number
2011-09-09T12:22:15.176086800Z - ERROR - FusionEventDriver - Invalid
FIO_AFFINITY parameter syntax at character 13: "5,g0,0xf;6,0x". Manual
affinity settings are disabled!
```

付録 J – VSL 2.x から 3.x へのデバイスのアップグレード

このバージョンの ioMemory VSL では、最新の ioMemory アーキテクチャや Flashback Protection 機能の向上など、新しい機能がサポートされています。これらの機能を使用するには、最新バージョンの Fusion-io ファームウェアが必要です。3.1.x 以降を実行するシステム内のすべての ioMemory デバイスは、最新バージョンのファームウェアにアップグレードする必要があります。

たとえば、ioMemory VSL 2.3.1 を実行しており、以前に ioDrive デバイスが取り付けられているシステムに新しい ioDrive2 デバイス(ioDrive2 デバイスには最新バージョンのファームウェアが必要)を取り付ける場合、既存のすべてのデバイスを最新のファームウェア バージョンにアップグレードする必要があります。

デバイスをアップグレードした後は、デバイスのファームウェアを以前のバージョンに戻すことはできません(以前のバージョンに戻すと、保証が無効となります)。

(VSL 2.x.x 用に構成されていた)デバイスをアップグレードして VSL 3.x.x に対応させるには、デバイスのローレベル メディア フォーマットが必要です。フォーマットすると、ユーザー データは消去されます。指示に従って、必ずすべてのデータをバックアップしてください。

アップグレード パス

デバイスの現在のファームウェア バージョンによっては、内部構造を維持するために、デバイスのファームウェアのアップグレードが複数回必要になることがあります。次に、最低限実行する必要があるアップグレード パスを示します。システム上の ioMemory VSL ソフトウェアは、この順序でアップグレードします(ソフトウェアの各バージョンと互換性のあるバージョンに**ファームウェアもアップグレードします**)。

1.2.4→1.2.7→2.1.0→2.3.1→3.x

たとえば、デバイスで ioMemory VSL バージョン 2.2.0 用のファームウェアを使用している場合は、(ioMemory VSL **および互換性のあるファームウェア**の両方を)2.3.1 にアップグレードしてから、パスに従ってアップグレードします。必要なすべてのソフトウェア バージョンおよびファームウェア バージョンの一覧については、**PRIMERGY ダウンロードページ:** **http://jp.fujitsu.com/platform/server/primergy/downloads/**を参照してください。

アップグレード手順

必ず上記のアップグレード パスに従ってください。以前に取り付けられているすべての ioDrive デバイスが、2.3.1 に互換性のある適切なファームウェアに更新されていることを確認します。

同じホストで ioDrive デバイスと ioDrive2 デバイスを使用する場合は、新しい ioDrive2 デバイスを取り付ける**前に**既存のすべての ioDrive デバイスでこのアップグレードを実行します。

1. 既存の各 ioDrive デバイスでアップグレードの準備を行います。
    1. 各 ioDrive デバイス上のユーザー データをバックアップします。

        アップグレードを行うには、デバイスのローレベル メディア フォーマットが必要です。フォーマットすると、ユーザー データは消去されます。すべてのデータをバックアップしてください。

    2. 任意のバックアップ方法を使用します。過去に実績があるソフトウェアおよびバックアップ デバイスを使用することをお勧めします。同じシステム上の別の ioMemory デバイスにはデータをバックアップしないでください。ローカル ディスク、または外部接続ボリュームにバックアップする必要があります。
    3. fio-bugreport コマンドライン ユーティリティを実行して、出力を保存します。これにより、システム内の各デバイスのデバイス情報がキャプチャされます。このデバイス情報は、アップグレードの問題をトラブルシューティングする場合に役立ちます。コマンド例:

        ```
        fio-bugreport
        ```

    4. ioDrive デバイスを切断します。次に例を示します。

        ```
        fio-detach /dev/fct*
        ```

        詳細については、「fio-detach」を参照してください。

2. 現在の ioMemory VSL ドライバーをアンロードします。次に例を示します。

    ```
    $ modprobe -r iomemory-vsl
    ```

    1. ioMemory VSL 2.x ソフトウェアをアンインストールします。
    2. [スタート]ボタン、[コントロール パネル]の順にクリックします。
    3. [プログラムと機能]をクリックします。
    4. ioMemory VSL（Fusion-io）のエントリをクリックします。
    5. [アンインストール]をクリックします。
    6. コンピューターを再起動します。
3. 新しい VSL をインストールします。

1. **PRIMERGY ダウンロードページ:** http://jp.fujitsu.com/platform/server/primergy/downloads/から Windows 用の ioMemory VSL インストール プログラムをダウンロードします。
2. ioMemory VSL インストール プログラムを実行します。インストール プログラムでは、インストールのオプションを示すカスタム セットアップ ツリービューが表示されます。

   ツリー ビューのコンポーネント上にマウス ポインターを合わせると、説明が表示されます。

3. [Next]をクリックします。
4. インストール用に異なるフォルダーを選択するには、フォルダーを参照して、[OK]をクリックします。デフォルトのフォルダーは `C:\Program Files\Fusion-io ioMemory VSL` です。

   アンインストーラー ファイルは、Fusion-io ioMemory VSL(デフォルトのインストール フォルダー)のルートに配置されます。

5. 画面上のプロンプトに従って、インストールを完了します。
6. インストーラーの完了画面で、[Reboot Now]をクリックします。

   すべてのインストール手順については、「既存の ioMemory VSL インストール」を参照してください。

4. `fio-update-iodrive` を使用して、各デバイスのファームウェアを最新バージョンに更新します。

   停電の回避
   更新中に停電が発生しないように、UPS などの保護手段を用意してください。更新中に停電が起こると、デバイスで障害が発生する可能性があります。このユーティリティに関するすべての警告、アラート、およびオプションについては、付録の `fio-update-iodrive` ユーティリティ リファレンスを参照してください。

5. サンプルの構文:
6. `fio-update-iodrive <iodrive バージョン.fff>`
7. `<iodrive バージョン.fff>`は、ファームウェア アーカイブへのパスです(デフォルトのファイル パスは `C:\Program Files\Fusion-io ioMemory VSL\Firmware\iodrive_<バージョン>.fff` です)。このコマンドによって、すべてのデバイスが選択されたファームウェアに更新されます。特定のデバイスを更新する場合のオプションについては、ユーティリティ リファレンスを参照してください。
8. システムを再起動します。

   `fio-status` を実行すると、アップグレードされたデバイスに lebmap がないという警告が表

示されます。これは予期された動作です。次の手順で修正されます。

データの消去を伴う手順
次の手順で `fio-format` を実行すると、ユーザー データを含むデバイス全体が消去されます。このフォーマットを開始した後は、デバイスを 2.x のドライバーにダウングレードすることはできません。ダウングレードすると、保証が無効となります。

9. `fio-format` を使用して、各デバイスをフォーマットします。次に例を示します。

   ```
   fio-format <デバイス>
   ```

   デバイス上のすべてのデータを消去するかどうかを確認するプロンプトが表示されます。

   デバイスの消耗度によっては、フォーマットの実行に長い時間がかかることがあります。

10. すべての ioDrive デバイスを接続します。次に例を示します。

    ```
    fio-attach /dev/fct*
    ```

11. fio-status を使用して、すべてのデバイスのステータスを確認します。

    ```
    fio-status -a
    ```

これで、ioDrive デバイスがこのバージョンの ioMemory VSL 用に正常にアップグレードされました。ioDrive2 デバイスを取り付けることができます。

付録 K – 付属資料の使用許諾

ドライバーにバイナリ形式で付属する `AVR bootloader` および `tree.h` ファイルには、付属資料に関する次の著作権要件を有するコンテンツが含まれています。

AVR Bootloader

Copyright © 2002, 2003, 2004, 2005, 2006, 2007 Eric B. Weddington

All rights reserved.

改変の有無にかかわらず、次の条件をすべて満たす場合に限り、ソースおよびバイナリ形式での本ソフトウェアの再頒布及び使用を許可します。

- ソース コードの再頒布は、上記の著作権情報、本条件一覧、および次の免責条項を保持して行う必要があります。
- バイナリ形式で再頒布する場合は、頒布時に提供される文書またはその他の資料に上記の著作権情報、本条件一覧、および次の免責条項を再表示する必要があります。
- 書面による事前の許可を得た場合を除き、本ソフトウェアから派生する製品の推奨または販売促進のために著作権保持者または寄与者の名前を使用することはできません。

本ソフトウェアは著作権保持者および寄与者によって「現状のまま」提供され、商品性および特定の目的への適合性に関する暗黙の保証を含め、いかなる明示的保証または暗黙的保証も行いません。著作権保持者または寄与者は、事由のいかんを問わず、損害発生の原因いかんを問わず、かつ責任の根拠が契約であるか厳格責任であるか（過失その他の）不法行為であるかを問わず、仮にそのような損害が発生する可能性を知らされていたとしても、本ソフトウェアの使用によって発生した（代替品または代用サービスの調達、使用の喪失、データの喪失、利益の喪失、業務の中断も含め、またそれに限定されない）直接損害、間接損害、偶発的損害、特別損害、懲罰的損害、または必然的損害について、一切責任を負わないものとします。

tree.h

Copyright © 2002 Niels Provos <provos@citi.umich.edu>
All rights reserved.

改変の有無にかかわらず、次の条件をすべて満たす場合に限り、ソースおよびバイナリ形式での本ソフトウェアの再頒布及び使用を許可します。

1. ソース コードの再頒布は、上記の著作権情報、本条件一覧、および次の免責条項を保持して行う必要があります。
2. バイナリ形式で再頒布する場合は、頒布時に提供される文書またはその他の資料に上記の著作権情報、本条件一覧、および次の免責条項を再表示する必要があります。

本ソフトウェアは作成者によって「現状のまま」提供され、商品性および特定の目的への適合性に関する暗黙の保証を含め、いかなる明示的保証または暗黙的保証も行いません。作成者は、事由のいかんを問わず、損害発生の原因いかんを問わず、かつ責任の根拠が契約であるか厳格責任であるか（過失その他の）不法行為であるかを問わず、仮にそのような損害が発生する可能性を知らされていたとしても、本ソフトウェアの使用によって発生した（代替品または代用サービスの調達、使用の喪失、データの喪失、利益の喪失、業務の中断も含め、またそれに限定されない）直接損害、間接損害、偶発的損害、特別損害、懲罰的損害、または必然的損害について、一切責任を負わないものとします。