



PartitionMagic Pro のスクリプト

以下の項目をクリックすると項目に対応する情報が表示されます。

- スクリプト処理の使用
- スクリプトの構文
- スクリプトコマンド
- スクリプトファイルステートメント
- 演算子
- クエリー
- スクリプトに関する注意事項
- サンプルスクリプト

スクリプト処理の使用

PowerQuest® PartitionMagic® Pro スクリプトを使用してコンピュータ上のパーティションとボリュームを変更できます。スクリプトとは、実行する処理操作を定義したテキストステートメントが記述されている ASCII テキストファイルです。ScriptBuilder またはテキストエディタを使用して、スクリプトファイルを作成できます。

新しいスクリプトの作成

1. PartitionMagic で、[ツール]→[スクリプト]→[ScriptBuilder] をクリックします。



2. ScriptBuilder で、[ファイル]→[新規] をクリックします。
3. [挿入] をクリックして、コマンド、クエリー、ステートメントのいずれかを選択します。
挿入する一部のコマンドにもパラメータを指定できます。

たとえば、Create コマンド (Create /FS=FAT など) をスクリプト編集ペインに挿入する場合は、[挿入]→[コマンド]→[作成] をクリックして、メニューからオプションを選択します。ScriptBuilderにある多くのコマンドの中でも Create コマンドは関連するさまざまなパラメータを持つため、挿入されるコマンドで指定されるよりも多くの情報を指定できます。この例では、Create コマンドはパーティションのラベル、サイズ、およびハードディスク上の位置を指定するパラメータを持ちます。

したがって、スクリプト編集ペインに挿入されているコマンドにパラメータを追加するには、まず追加したいコマンドの行をクリックし、次に[挿入]→[パラメータ]をクリックして、パラメータオプションを選択します。[挿入]をクリックしたとき[パラメータ]オプションが淡色表示(使用不可)であれば、そのコマンドにはパラメータがありません。

コマンド中の間違ったパラメータを置き換えるには、まずその間違ったパラメータを選択(強調表示)し、次に[挿入]→[パラメータ]をクリックして、新しいパラメータを選択します。

4. スクリプト編集ペインで適切なオプションを指定します。
5. スクリプトの作成が終わるまで手順3と4を繰り返します。
6. スクリプトファイルを保存するには、[ファイル]→[名前を付けて保存]をクリックし、ファイルのパスとファイル名を指定して、[OK]をクリックします。

スクリプトのテキストファイルには、C++形式(//)のコメントを記入できます。コメントはステートメントと別の行にするか、有効なステートメントの後に記述します。

構文を削除するには、削除したい行をスクリプト編集ペインで選択して、Delete キーを押します。スクリプト編集ペイン内で右クリックして、スクリプトから行を切り取り、コピー、貼り付け、削除することもできます。

スクリプトファイルの作成と保存が終わったら、いつでも[スクリプト]プルダウンメニューの[実行]オプションを使ってファイルをコンパイルできます。スクリプトをまだ保存していない場合、[実行]コマンドは[保存して実行]に変わります。

スクリプト構文のチェック

[構文チェック]は、スクリプト中の構文全体をテストして構文エラーの有無を調べるときに使います。

1. ScriptBuilder で、[スクリプト]→[構文チェック]をクリックします。

構文中のエラーは、スクリプト編集ペインのすぐ下にあるエラーペインにリスト表示されます。いずれかのペインの表示領域を拡大または縮小するには、水平分割線を上下にドラッグします。

スクリプトの記録

ScriptBuilder の[挿入]メニューで項目を選択してスクリプトを作成する代わりに、PartitionMagic を使いながらスクリプトを「記録」できます。

1. PartitionMagic で、[ツール]→[スクリプト]→[記録]をクリックします。
2. 通常どおりに PartitionMagic で操作を選択します。

スクリプトの記録中に一部の操作が淡色表示になり使用できない場合があります。淡色表示の操作をスクリプトに記録することはできません。

オプションを選択すると、関連するスクリプトコマンドがScriptBuilderに送信され、スクリプト編集ペインの現在のカーソル位置に挿入されます。

3. 記録を終了したら、[ツール]→[スクリプト]→[記録]をクリックして、記録をオフにします。
4. スクリプトの記録中にマシンに加えた仮想的な変更を破棄するかどうかを確認する画面が表示される場合があります。その場合、[はい]をクリックしてすべての変更を無効にするか、[いいえ]をクリックして変更をすべて適用します。
[はい][いいえ]のいずれをクリックしても、スクリプトに影響はありません。
5. [ファイル]→[保存]または[ファイル]→[名前を付けて保存]をクリックして、スクリプトファイルを保存します。

スクリプトの実行

ScriptBuilder、PartitionMagic Pro、またはPartitionMagic Proのコマンドラインからスクリプトを実行できます。

実行場所	操作手順
ScriptBuilder	[スクリプト]→[実行]または[保存して実行]をクリックします。
PartitionMagic Pro	[ツール]→[スクリプト]→[スクリプトファイルの実行]をクリックして、実行したい.PQSファイルのパスとファイル名を指定したら、[実行]をクリックします。
PartitionMagic Proのコマンドライン	/CMD コマンドラインスイッチを使って、PartitionMagicを開かないでコマンドラインからスクリプトを実行します。たとえば、スクリプトファイル名がSCRIPT.TXTの場合、スクリプトからプログラムを実行する構文は次のようになります。 PQMAGIC /CMD=SCRIPT.TXT

プログラムエラーが発生した場合、スクリプトはすぐに終了し、メッセージペインにエラーが表示されます。ログファイルまたはエラーファイルを見ることでも、エラーが発生したかどうかとそのエラーの種類を判断できます。この2つのファイルが作成されるのは、コマンドラインでそのように指定した場合のみです。

ディスクの環境設定がスクリプトコマンド用に正しく設定されていないため、スクリプトは操作を完了できないことがあります。そのような場合、“操作が残っています。”と“スクリプトが完了しました。”を報告します。エラーは報告されません。

スクリプトに記述された各操作は、最後に指定したパーティションで実行されます。スクリプトを実行する前に、適切なパーティションを指定する必要があります。

スクリプトの構文

スクリプトファイルステートメントの構文で使われる特殊文字の説明を以下に示します。スクリプトファイルステートメントでは、大文字と小文字を区別しません。操作する前に、対象のドライブとパーティションを選択する必要があります。

{ } - 必須のパラメータを意味します。

[] - オプションのパラメータを意味します。

| - 2つ以上のオプションの中からの選択することを意味します。

スクリプトコマンド

スクリプトコマンドを使って、PartitionMagic の [操作] メニューから利用できるほとんどすべての操作を実行できます。

[挿入] → [コマンド] をクリックして、コマンドを ScriptBuilder のスクリプト編集ペインに挿入します。

スクリプト編集ペインに挿入されているコマンドに省略可能なまたは必須のパラメータを追加するには、まず追加したいコマンドの行をクリックし、次に [挿入] → [パラメータ] をクリックして、パラメータを選択します。[挿入] をクリックしたとき [パラメータ] オプションが淡色表示 (使用不可) であれば、そのコマンドにはパラメータがありません。

Allow Manual Reboot

変更が行われた後、プログラムがコンピュータをリモートで再起動できなくなった場合でも、スクリプトの実行を可能にします。使用された場合の最初のステートメントは、次のとおりです。

パラメータはありません。

Bad Sector Retest

現在のパーティションで不良セクタを再診断して、不正に設定された不良セクタのマークを外します。

Check

選択されたパーティションにエラーがないかどうか調べるのに使用します。

パラメータはありません。

ほとんどすべてのスクリプトで Check コマンドを使って、影響している各パーティションを診断する必要があります。ディスクエラーが存在する場合、Check コマンドはそのエラーを検出し、問題が起こる前にスクリプトを終了します。

Cluster Analyzer

構文

```
Cluster Analyzer [/SetClusterSize={ Recommended|512 | 1 | 2 | 4  
| 8 | 16 | 32 | 64 }]
```

特定のパーティションについてのクラスタアナライザ情報を取得します。

パラメータなしでこのオプションが使用されると、クラスタアナライザ画面が表示されます。

パラメータ	説明
/SetClusterSize	クラスタを指定のサイズに変更します。

Convert To FAT

FAT32 または NTFS パーティションを FAT に変換します。

Convert To FAT32

FAT または NTFS パーティションを FAT32 に変換します。

Convert To HPFS

FAT パーティションを HPFS に変換します。

Convert to NTFS

FAT パーティションを NTFS に変換します。Windows® 2000 では、FAT32 パーティションを NTFS パーティションに変換できます。このコマンドは、Windows で使用する必要があります。

Convert To Primary

論理パーティションを基本パーティションに変換します。

Convert To Logical

基本パーティションを論理パーティションに変換します。

Copy

構文

```
Copy [/Position = {Beginning | End}]
```

パラメータ	説明
/Position	(必要に応じて) 空き領域でパーティションがコピーされる位置を指定するためにENDまたはBEGINNINGを記入する必要があります。デフォルトはBeginningです。

パーティションを空き領域にコピーします。Copy コマンドを正しく実行するには、コピー元のディスクとパーティションを選択し、コピー先のディスクと空き領域を選択する必要があります。

Copy コマンドは、次のコマンドより後に記述してください。

- Select Disk {Num}
- Select Partition {PartitionLetter | "Volume Label" | Extended | Next | Previous | Num }
- Select Destination Disk {Num}
- Select Destination Unallocated {Num | First | Last | Largest | After Partition Num | Before Partition Num | Next | Previous}

Create

構文

```
Create /FS={FAT | FAT32 | HPFS | LINUXEXT2 | LINUXSWAP | NTFS |  
EXTENDED | UNFORMATTED} [/Label="NEW LABEL"] [/Size=Value]  
[/Position={BEGINNING | END}]
```

新しいパーティションを作成します。オプションで指定していれば初期化します。

パラメータ	説明
/FS	(必須) 上で指定した文字列のいずれか。特定の/FSを指定した場合に作成が失敗することがあります。たとえば、既に存在するのに拡張パーティションを作成しようとした場合などです。

パラメータ	説明
/Label	(必要に応じて)「NEW LABEL」を適切なボリュームラベルで置き換えます。FAT パーティションの場合は 11 文字以下にしてください。 Linux Ext2 パーティションのラベルは 16 文字以下に、NTFS パーティションの場合は 32 文字以下にしてください。ラベルは二重引用符で囲む必要があります。無効な文字が入力されると、スクリプトに障害が発生することがあります。※2 バイト文字列を利用することはできません。
/Size	(必要に応じて)メガバイト単位で指定します。指定なしの場合、デフォルトで空き領域のサイズになります。
/Position	(必要に応じて)空き領域でパーティションが作成される位置を指定するためにENDまたはBEGINNINGを記入する必要があります。デフォルトはBEGINNINGです。

Delete

構文

```
Delete {Volume Name | "NO NAME" | "SWAPSPACE2" | "UNKNOWN"} [/Shred]
```

選択したパーティションを削除するために使います。

パラメータ	説明
Volume Name	ボリューム名を入力して、指定したパーティションのデータのみが破棄されるようにする必要があります。入力する値は常に二重引用符で囲む必要があります。パーティションラベルが空白で、パーティションがFATまたはHPFSの場合は、「"NO NAME"」と入力します。パーティションがFATまたはHPFSでない場合は、「"UNKNOWN"」と入力します。初期化されていないパーティションまたは拡張パーティションを削除するには、「"NO NAME"」と入力します。Linuxのスワップパーティションを削除するには、「"SWAPSPACE2"」と入力します。
/Shred	(必要に応じて)データと選択されたパーティションを完全に破棄します。Undeleteは、このコマンドを取り消すためには使用できません。このコマンドはDeleteよりも時間が長くなります。 Shredパラメータは、PartitionMagicの[削除してシュレッド]操作と同じ機能を持ちます。

Delete All

構文

```
Delete All [/Shred]
```

選択したディスクからすべてのパーティションを削除するために使います。

Format

構文

```
Format {Volume Name | "NO NAME" | "UNKNOWN" } /FS={ FAT | FAT32  
| HPFS | LINUXEXT2 | LINUXSWAP | NTFS } [/Label="NEW LABEL"]
```

選択したパーティションを初期化するために使います。

パラメータ	説明
Volume Name	拡張パーティションまたは未初期化パーティション以外のパーティションを初期化するために必要です。これは、適切なパーティションのデータのみを破棄しようとしていることを確認するためです。入力する値は常に二重引用符で囲む必要があります。パーティションラベルが空白でパーティションがFATかHPFSの場合は、ラベルに「"NO NAME"」と入力します。パーティションがFATかHPFS以外の場合は、ラベルに「"UNKNOWN"」と入力します。※2バイト文字列を含むラベル名は指定できません。
/FS	(必須)上で指定した文字列のいずれか。特定の/FSを指定した場合に作成が失敗することがあります。たとえば、1024シリンダを超えるFATパーティションを初期化しようとした場合などです。
/Label	(必要に応じて)「NEW LABEL」を適切なボリュームラベルで置き換えます。FATパーティションの場合は11文字以下にしてください。Linux Ext2パーティションのラベルは16文字以下に、NTFSパーティションの場合は32文字以下にしてください。ラベルは二重引用符で囲む必要があります。無効な文字が入力されると、スクリプトに障害が発生することがあります。※2バイト文字列を利用することはできません。

Hide

現在選択されているパーティションを非表示にします。「Unhide」を参照してください。

Info

構文

Info [/Usage] [/Waste] [/Partition] [/FS]

選択したパーティションについての情報を表示するときに使います。すべてのパラメータは省略可能です。パラメータが指定されていない場合は、選択したパーティションのすべての情報を表示します。

パラメータ	説明
/Usage	<p>[ディスクの使用状況] ページは、FAT、FAT32、NTFS、およびHPFS ファイルシステムに関してのみ表示されます。</p> <p>この画面には、次の情報がバイト単位、メガバイト単位、およびパーセント単位で表示されます。</p> <ul style="list-style-type: none">パーティション内の使用済み領域 (クラスタによる浪費領域を含む)パーティション内の未使用領域パーティション内の不良領域パーティション内の合計容量 (前述の3項目の合計)
/Waste	<p>[浪費クラスタ] ページは、FAT および FAT32 ファイルシステムを使用しているパーティションに関してのみ表示されます。この画面では次の情報が表示されます。</p> <ul style="list-style-type: none">現在のクラスタサイズのバイト数またはキロバイト数パーティションに格納されているデータ (バイト単位およびメガバイト単位で表示)パーティション内の浪費領域 (バイト単位およびメガバイト単位で表示) <p>使用済み領域の合計バイト数とメガバイト数 (前述の2項目の合計)</p>
/Partition	<p>[パーティション情報] 画面は、すべての種類のパーティションに関して表示されます。これには、空き領域と拡張パーティションも含まれます。このページには次の情報が表示されます。</p> <ul style="list-style-type: none">[パーティションの種別] には、パーティションやファイルシステムの種別 (FAT、FAT32、NTFS、HPFS など) が16進数とテキストで表示されます。16進数による表示は、パーティション種別を表す一般的な方法です。[シリアル番号] の表示があるのは、パーティションのファイルシステムがシリアル番号を使っている場合です。すべてのファイルシステムがシリアル番号を使うわけではありません。

パラメータ	説明
-------	----

/Partition の続き	<p>パーティションについての物理情報を表示します。</p> <ul style="list-style-type: none">• [先頭物理セクタ]には、パーティション開始位置の論理番号と場所(シリンダ、ヘッド、およびセクタ番地)が表示されます。• [最終物理セクタ]には、パーティション終了位置の論理番号と場所(シリンダ、ヘッド、およびセクタ番地)が表示されます。• [物理セクタ総数]は、パーティション内のセクタ数を表示します。• [物理ジオメトリ]は、パーティションが置かれている物理ディスクドライブのシリンダ数、ヘッド数、セクタ数を表示します。
----------------	---

/FS FAT または FAT32	<p>FAT パーティションの場合、このパラメータは次の情報の画面を表示します。このページの上段には、ファイルシステムについて次の情報が表示されます。</p> <ul style="list-style-type: none">• [セクタ/FAT]• [ルートディレクトリ容量]• [先頭FATセクタ]• [先頭データセクタ]
-------------------------	---

このページの下段には、以下の情報が表示されます。

- パーティションにあるファイルのバイト数、ファイル数、および非表示ファイルの数。
- パーティション内にあるディレクトリのバイト数、ディレクトリ数、および非表示ディレクトリの数。

FAT ファイルシステムに対する拡張機能がいくつかあります。このページの下段には、FAT 拡張機能に関する次の情報が表示されます。

- OS/2 拡張属性で使用されているバイト数、および拡張属性の影響を受けるファイル数とディレクトリ数。

/FS NTFS	<p>NTFS パーティションの場合、このパラメータは次の情報の画面を表示します。この画面は、選択されたパーティションについて次のファイルシステム情報を表示します。</p>
-------------	--

- **[NTFS バージョン]**は、NTFS のバージョン番号を表示します。
- **[バイト/NTFSセクタ]**は、選択されたパーティションでの各論理セクタのバイト数を表示します。物理セクタ単位のバイト数は常に512です。
- **[クラスタサイズ]**
- **[先頭MFTクラスタ]**
- **[ファイルレコードのサイズ]**

パラメータ	説明
/FS NTFS の続き	<p>下段に表示されている情報は、次の情報を含んだNTFS CHKDSK でレポートされる内容と似ています。</p> <ul style="list-style-type: none"> ファイル数およびこれらに割り当てられているバイト数とクラスタ数。 ファイルで使用されているクラスタ内で、クラスタのサイズが原因で浪費されているバイト数。 インデックス(ディレクトリ)の数と、これらに割り当てられている領域のバイト数とクラスタ数。 <p>ほかのシステム構造体で使用するために確保されているバイト数とクラスタ数。</p>
/FS HPFS	<p>HPFS パーティションの場合、このパラメータは次の情報の画面を表示します。このページの上段には、ファイルシステムについて次の情報が表示されます。</p> <ul style="list-style-type: none"> [パーティション状況](パーティションがアクティブかどうか) [ディレクトリブロック セクタ] [空きディレクトリブロック] [使用済 Hot Fix] <p>このページの下段には、以下の情報が表示されます。</p> <ul style="list-style-type: none"> パーティション内にあるファイルのバイト数、ファイル数、およびそれに対応するセクタ数。 ファイルセクタで未使用のバイト数 パーティション内にあるディレクトリのバイト数、ディレクトリ数、およびそれに対応するセクタ数。 file/dir Fnode のバイト数とそれに対応するセクタ数。 システム用に確保されているバイト数とそれに対応するセクタ。 拡張属性のバイト数。

Label

構文

```
Label [/GetLabel ] [/SetLabel="NAME"]
```

パラメータを指定しない場合のデフォルトはGetLabelであり、選択したパーティションのラベルを報告します。

/SetLabel オプションによって、選択したパーティションの名前を変更できます。ラベルは、FAT、FAT32、および HPFS パーティションでは 11 文字以下、NTFS パーティションでは 32 文字以下、Linux Ext2 パーティションでは 16 文字以下にしてください。FAT、FAT32、または HPFS パーティション種別のラベルの命名規則は、DOS での命名規則とほぼ同じですが、例外が 2 つあります。まず、スペースが使用できます。そして、最初の 8 文字と最後の 3 文字の間にピリオドは必要ありません。

Merge

構文

```
Merge /Target={First | Second} /Folder="NAME" [/FS={FAT | FAT32 | NTFS}]
```

Merge は、ハードディスク上で互いに隣接する 2 つの FAT または FAT32 パーティションを結合するときに使います。2 つのパーティション間には空き領域が存在してもかまいません。これは、ディスク上の最大パーティション数に達したが、パーティションを削除したくない場合に役立ちます。

Merge を使って、同じクラスタサイズと同じバージョン種別を持つ 2 つの隣接する NTFS パーティションを結合することもできます。2 つの NTFS パーティションが同じクラスタサイズと同じバージョン種別を持つことを確認するには、パーティションマップ中のパーティションを右クリックし、メニューから **[情報]** を選択します。**[NTFS 情報]** タブをクリックし、クラスタサイズとバージョン種別を表示します。

Merge コマンドが正しく機能するには、隣接する 2 つの FAT または FAT32 パーティションを選択するか、同じクラスタサイズと同じバージョン種別を持つ 2 つの NTFS パーティションを選択する必要があります。オペレーティングシステムを含むパーティションを結合することは避けてください。

Merge コマンドは、次のコマンドより後に記述する必要があります。

- Select Disk {Num}
- Select Partition {{Number} | {PartitionLetter} | {"Volume Label"} | First | Last | Next | Previous | Extended}
- Select Merge Partition {Next | Previous}

パラメータ	説明
/Target	保持したいパーティションを指定します。たとえば、First を選択した場合、選択した 2 番目のパーティションは最初のパーティションに結合されます。

パラメータ	説明
/Folder	対象のパーティションに結合されるデータのフォルダ名を指定します。たとえば、DATA パーティションと BACKUP パーティションがあつて、DATA パーティションを保持する場合、フォルダ名に「“BACKUP”」を指定します。結合後、BACKUP パーティションのすべてのデータは、DATA パーティション内の BACKUP フォルダに存在します。
/FS= FAT FAT32 NTFS	(必要に応じて)作成されるファイルシステムを指定できます。このパラメータを使用しなければ、PartitionMagic Pro は最適なファイルシステムを自動的に選択します。

Move Left

構文

Move Left {Max | Min | Value }

パーティションを左へ移動します。

パラメータ	説明
Max	パーティションをできる限り左に移動します。前のパーティションまたはドライブの先頭との間にすきまがなくなるようにします。
Min	パーティションを最小値分(1 シリンダ)だけ左に移動します。
Value	メガバイト単位で指定された値だけパーティションを左に移動します。

Move Right

構文

Move Right {Max | Min | Value }

パーティションを右に移動します。

パラメータ	説明
Max	パーティションをできる限り右に移動します。次のパーティションまたはドライブの最後との間にすきまがなくなるようにします。
Min	パーティションを最小値分(1 シリンダ)だけ右に移動します。

パラメータ	説明
Value	メガバイト単位で指定された値だけパーティションを右に移動します。

Move Space After

構文

Move Space After {Max | Value }

Move Left および Move Right と同じです。

パラメータ	説明
Max	Move Left Max と同じです。パーティションを左に移動してパーティションの後の領域をできる限り大きくします。
Value	可能であれば、後の領域がメガバイト単位で指定した値に等しくなるようにパーティションを左または右に移動します。

Move Space Before

構文

Move Space Before {Max | Value }

Move Right と同じです。

パラメータ	説明
Max	Move Right Max と同じです。パーティションを右に移動してパーティションの前の領域をできる限り大きくします。
Value	前の領域がメガバイトで指定した値と同じになるように、パーティションを右に移動します。

Resize

構文

Resize {Max | Min | Value } [/SetClusterSize={512 | 1 | 2 | 4 | 8 | 16 | 32 | 64}]

パーティションのサイズを変更します。

パラメータ	説明
Max	できる限り大きなサイズに変更します。パーティションの右端はドライブの最後、または可能であれば次のパーティションにぴったりとつけます。
Min	できる限り小さいサイズに変更します。データのサイズによって異なります。
Value	メガバイト単位で指定した値にサイズ変更します。
/SetClusterSize	クラスタサイズを、サイズ変更操作中に指定したサイズに設定します。クラスタサイズは、指定されたパーティションサイズに対して有効なサイズである必要があります。

Resize Larger

構文

```
Resize Larger {Max | Min | Value} [/SetClusterSize={512 | 1 | 2 | 4 | 8 | 16 | 32 | 64}]
```

増分を指定してパーティションのサイズを増加させます。「**Resize**」を参照してください。

パラメータ	説明
Max	Resize Max と同じです。パーティション領域を最大にします。
Min	パーティションのサイズを最小値分(1 シリンダ)だけ大きくします。
Value	メガバイト単位で指定したサイズだけパーティションのサイズを大きくします。
/SetClusterSize	サイズ変更操作中に、指定したサイズにクラスタを設定します。クラスタのサイズは指定されたパーティションのサイズと対応している必要があります。

Resize Left Boundary

構文

```
Resize Left Boundary {Max | Min | Value }
```

左の境界を移動して拡張パーティションのサイズを変更します。この操作は拡張パーティションだけで利用可能です。

パラメータ	説明
Max	できる限り大きなサイズに変更します。パーティションの左端はドライブの左端、または可能であれば前のパーティションにぴったりとつけます。
Min	できる限り小さいサイズに変更します。データのサイズによって異なります。
Value	メガバイト単位で指定した値にサイズ変更します。

Resize Left Boundary Larger

構文

```
Resize Left Boundary Larger {Max | Min | Value }
```

拡張パーティションのサイズを左の境界位置を指定して増大させます。この操作は拡張パーティションだけで利用可能です。「Resize Left Boundary」も参照してください。

パラメータ	説明
Max	Resize Left Boundary Max と同じです。パーティション領域を最大にします。
Min	パーティションのサイズを最小値分(1 シリンダ)だけ大きくします。
Value	メガバイト単位で指定したサイズだけパーティションのサイズを大きくします。

Resize Left Boundary Smaller

構文

```
Resize Left Boundary Smaller {Max | Min | Value }
```

拡張パーティションのサイズを左の境界位置を指定して減少させます。この操作は拡張パーティションだけで利用可能です。「Resize Left Boundary」も参照してください。

パラメータ	説明
Max	Resize Left Boundary Min と同じです。パーティション領域を最小にします。

パラメータ	説明
Min	パーティションのサイズを最小可能量(1 シリンダ)だけ小さくします。
Value	メガバイト単位で指定したサイズだけパーティションのサイズを小さくします。

Resize Smaller

構文

```
Resize Smaller {Max | Min | Value}
[/SetClusterSize={512 | 1 | 2 | 4 | 8 | 16 | 32 | 64 }]
```

増分を指定してパーティションのサイズを減少させます。「Resize」も参照してください。

パラメータ	説明
Max	Resize Min と同じです。パーティション領域を最小にします。
Min	パーティションのサイズを最小可能量(1 シリンダ)だけ小さくします。
Value	メガバイト単位で指定したサイズだけパーティションのサイズを小さくします。
/SetClusterSize	サイズ変更操作中、クラスタサイズに指定したサイズを設定します。クラスタサイズは、指定されたパーティションサイズに対して有効なサイズである必要があります。

Resize Space After

構文

```
Resize Space After {Max | Min | Value}
[/SetClusterSize={512 | 1 | 2 | 4 | 8 | 16 | 32 | 64 }]
```

サイズ変更後、パーティションの後に続く空き領域を指定してパーティションをサイズ変更します。この操作は、拡張パーティションを含むすべてのパーティションに利用可能です。「Resize」も参照してください。

パラメータ	説明
Max	パーティションの後の空き領域ができる限り大きくなるようにサイズを変更します。パーティション領域は最小になります。

パラメータ	説明
Min	パーティションの後の空き領域ができる限り小さくなるようにサイズを変更します。パーティション領域は最大になります。
Value	後の領域がメガバイト単位で指定した値のサイズになるようにパーティションのサイズを決めます。
/SetClusterSize	サイズ変更操作中、クラスタサイズに指定したサイズを設定します。クラスタサイズは、指定されたパーティションサイズに対して有効なサイズである必要があります。

Resize Space Before

構文

Resize Space Before {Max | Min | Value }

サイズ変更終了後、パーティションの前の空き領域を指定して拡張パーティションのサイズを変更します。この操作は拡張パーティションだけで利用可能です。

パラメータ	説明
Max	パーティションの前の領域ができる限り大きくなるようにサイズを変更します。パーティション領域は最小になります。
Min	パーティションの前の領域ができる限り小さくなるようにサイズを変更します。パーティション領域は最大になります。
Value	前の領域がメガバイト単位で指定した値のサイズになるようにパーティションのサイズを決めます。

Resize Root

構文

Resize Root {Value | Min | Max }

FAT パーティションのルートディレクトリのエントリ数を変更します。

パラメータ	説明
Value	512 から 1024 までの値を指定します。これは、このパーティションに対して可能な最大ルートエントリ数を変更します。実際に設定される値は、指定した値の近似値となります。
Min	パーティションの最大ルートエントリ数を最小可能数に設定します。

パラメータ	説明
Max	ルートエントリの最大数が可能な最大値になるようにパーティションを設定します。

Select

文字またはボリュームでパーティションを選択する場合は、ディスクを選択する必要はありません。指定した文字またはボリュームが見つかるまで、すべてのディスクが順に検索されます。つまり、2つのパーティションのボリューム名が同じ場合は、先に見つかった方のパーティションが選択されます。

Next および Previous のコマンドは、同じ種類の直前の選択を基準にしています。Select Partition Next または Select Partition Previous を使う場合は、その前にパーティションを選択しておく必要があります。同様に、Select Unallocated Next または Select Unallocated Previous を使う前にも、空き領域を選択しておく必要があります。

Select Disk

構文

```
Select Disk {Number}
```

ハードディスクを選択するのに使用します。

パラメータはありません。

Select Partition

構文

```
Select Partition [{Number} | {Letter} | {"Volume"} | First | Last | Next | Previous | Extended}]
```

パーティションを選択します。パーティションを文字で選択する場合、文字を大文字で入力する必要があります。

パラメータはありません。

Select Unallocated

構文

```
Select Unallocated [{Number} | First | Last | Largest | Next | Previous | After Selected Partition | Before Selected Partition}]
```

ディスク上の空き領域を選択します。

パラメータはありません。

Select Merge Partition

構文

```
Select Merge Partition {Next | Previous}
```

隣接パーティションと結合するパーティションを選択します。このコマンドは **Select Partition** コマンドの後に記述する必要があります。

パラメータはありません。

Select Destination Disk {Number}

対象ハードディスクを選択します。このコマンドは **Copy** コマンドで使用されます。

パラメータはありません。

Select Destination Unallocated

構文

```
Select Destination Unallocated | {{Number} | First | Last |  
Largest | After Partition {Number} | Before Partition {Number} |  
Next | Previous}
```

パーティション内の対象空き領域を選択するのに使用します。このコマンドは **Copy** コマンドで使用されます。

パラメータはありません。

Set Active

選択したパーティションをアクティブ、すなわち起動可能なパーティションとして設定します。

パラメータはありません。

Set Allow User Cancel {ON | OFF}

構文

```
Set Allow User Cancel [ON | OFF]
```

Set Allow User Cancel OFF を設定すると、実行中のスクリプトは中断できません。このコマンドはデフォルトではオンになります。キャンセルボタンが淡色表示の場合、コマンドはオフになります。

このコマンドは、スクリプトの実行中にそのスクリプトを中断したくないときに使います。実行中のスクリプトを中断すると、ハードドライブを損傷する恐れがあります。

Set Default Bad Sector Test State { ON | OFF }

現在選択されているドライブのすべてのパーティションに対して不良セクタテストをオンまたはオフに設定します。"/BadSectorTest" オプションはこの設定を無効にします。

Set Drive Read Only Mode {ON | OFF}

現在選択しているドライブ上のすべてのパーティションについて、読み込み専用フラグをオンまたはオフに設定します。ドライブがオンに設定された場合、パーティションへの変更は許可されません。読み込み専用ドライブ上にある BOOT.INI ファイルに対しては、パーティションの追加、削除、またはコピーに伴う変更が一部許可されます。

Set Ignore OS/2 EA Errors { ON | OFF }

システムに OS/2 がいない場合、このオプションを ON にすると、FAT パーティションの診断時に OS/2 拡張属性のエラーを無視することができます。

Set NT 64K FAT Clusters { ON | OFF }

このオプションに ON が設定されている場合、64K クラスタの FAT パーティションを作成できます。また、PartitionMagic Pro では、最大 4GB の FAT パーティションを作成できます。

Windows NT または Windows 2000 では、64K クラスタだけを使う必要があります。ほかのオペレーティングシステムを使用する場合は、64K 以外のクラスタを使う必要があります。

Set Force User Logoff { ON | OFF }

ユーザーのログオフをオンまたはオフに設定します。オンに設定すると、スクリプトの残りが実行される前に、ユーザーは強制的にシステムからログオフされます。

Show

構文

Show {Partitions | Preference | Destination}

選択されたパラメータに関する要約を表示します。

パラメータ	説明
Partition	パーティションについての情報を表示します。ディスクを選択する必要があります。
Preference	[一般]→[環境設定]メニューの環境設定を表示します。
Destination	対象ディスクについての情報を表示します。このコマンドは Select Destination Disk コマンドの後に記述する必要があります。

Unhide

現在選択されているパーティションを非表示から表示に切り替えます。「Hide」を参照してください。

パラメータはありません。

スクリプトファイルステートメント

スクリプトステートメントでは、条件や変数に従ってスクリプトの流れを変更できます。ScriptBuilderで[挿入]→[ステートメント]をクリックすることで、ステートメントを挿入できます。

変数

指定された名前の変数を作成します。作成した変数には、後で条件式や数式で使うための値を割り当てることができます。

構文

```
Dim {変数名}
```

たとえば、次の行はNumberという名前の変数を作成し、そこに値2を割り当てます。

```
Dim Number  
Number = 2
```

If...End If

指定した条件に応じて異なるステートメントを実行します。それぞれのIf {} Then ステートメントの後にEnd Ifステートメントが続く必要があります。Else If {} ThenとElseのステートメントは省略可能です。

構文

```
If {条件} Then  
// ここにコマンドを記述  
Else If {条件} Then  
// ここにコマンドを記述  
Else  
// ここにコマンドを記述  
End If
```

Do...Loop While

条件が偽または0になるまで繰り返しステートメントを実行します。Do と Loop While {} コマンドの間にあるステートメントは、条件のテスト前に必ず1回実行されます。

構文

```
Do
// ここにコマンドを記述
Loop While { 条件 }
```

Do While...Loop

条件が偽または0になるまで繰り返しステートメントを実行します。Do While {} と Loop コマンドの間にあるステートメントは、最初にテストする前に条件が偽または0の場合は決して実行されません。

構文

```
Do While { 条件 }
// ここにコマンドを記述
Loop
```

Exit Loop

Exit Loop コマンドは、ループの最後に達する前に Do...Loop While または Do While...Loop ステートメントからジャンプするときに使います。

ループまたはスクリプトがエラーで停止した後にコマンドが存在する必要があります。

演算子

PartitionMagic はスクリプトで次の演算子をサポートします。

演算子	動作
条件	
<	左オペランドが右オペランドよりも小さいとき真を返します。
=	両オペランドが等しいとき真を返します。
>	左オペランドが右オペランドよりも大きいとき真を返します。
<=	左オペランドが右オペランドよりも小さいか等しいとき真を返します。

演算子	動作
>= または ==	左オペランドが右オペランドよりも大きいとき等しいとき真を返します。
<>	両オペランドが等しくないとき真を返します。
演算記号	
+	左右のオペランドを加算します。
-	左オペランドから右オペランドを減算します。
*	左右のオペランドを乗算します。
/	左オペランドを右オペランドで除算します。
代入	
=	右オペランドの値を左オペランドに代入します。たとえば、X=3 は、Xに3の値が設定されます。パラメータ内に演算子を使用することもできます。

スクリプト中の数値は真または偽として処理することができます。条件式(IFと両方のループ)が許されるステートメントには、数式(クエリーや算術式)を入力できます。
PartitionMagic Proのスクリプトでは、ゼロをFALSE、ゼロ以外をTRUEとして扱います。
たとえば、次の2つの行は共にTRUEまたはFALSEと評価し、同じ効果と結果を持ちます。

```
IF GetTotalPartitions then ...
IF GetTotalPartitions > 0 then ...
```

演算子の例

次の例は、スクリプト全体でさまざまな演算子を使います。

```
// PowerQuest PartitionMagic Proスクリプトファイル
Select Disk 2
Select Partition D
//2倍にサイズ変更する
Resize Larger GetSelectedPartitionSize * 2
// 未使用領域を半分にサイズ変更する
Resize Smaller GetUsedAmount / 2
// 未使用領域の半分を使ってパーティションを作成する
Select Unallocated Largest
Create /FS=FAT32 /Size=GetUnallocatedSize * 0.5
// 領域が残っていれば、300MBのパーティションを作成する
```

```

Dim Num
Num = 300
Select Unallocated Largest
If GetUnallocatedSize > Num Then
    Create /FS=FAT32 /Size=Num
End If

```

クエリー

ScriptBuilder でスクリプトにクエリーを挿入するには、[挿入]→[クエリー]をクリックします。クエリーはスクリプト編集ペインのカーソル位置に挿入され、選択済みテキストを置き換えます。

クエリーから返された値は、変数に割り当てたり、条件式や数式で使うことができます。次の表は使用できるクエリーを示します。

クエリー	使用目的と構文
Number of Disks	<p>選択したシステム上にある物理ディスクの合計数を表示します。</p> <p>構文: GetTotalDisks</p>
Number of Partitions	<p>選択したディスク上にあるパーティションの合計数を表示します。</p> <p>構文: GetTotalPartitions</p>
Number of Unallocated	<p>選択したディスク上にある空き領域の合計数を表示します。</p> <p>構文: GetTotalUnallocatedSpaces</p>
Disk Size	<p>現在選択しているディスクのサイズをメガバイト単位で表示します。</p> <p>構文: GetDiskSize</p>
Disk Allocated {Size Percentage}	<p>パーティションに割り当てられたディスクのサイズをメガバイト単位またはパーセント単位で表示します。</p> <p>構文: GetAllocatedSize GetAllocatedPercent</p>

クエリー	使用目的と構文
Disk Unallocated {Size Percentage}	<p>現在選択しているディスク上にある残りの空き領域のサイズをメガバイト単位またはパーセント単位で表示します。</p> <p>構文: GetUnallocatedSize GetUnallocatedPercent</p>
Partition Size	<p>パーティションのサイズをメガバイト単位で表示します。</p> <p>構文: GetSelectedPartitionSize</p>
Partition Number	<p>選択したパーティションのパーティション番号を表示します。返された値は、Select Partition {番号} コマンドで使えます。戻り値0は、そのパーティションが存在しないことを示します。たとえば、拡張パーティションがない場合、GetPartitionNumber Extended コマンドは0を返します。</p> <p>構文: GetPartitionNumber {<number> <Letter> "NAME" First Last Next Previous Extended }</p>
Partition Used	<p>パーティション上の使用済み領域のサイズをメガバイト単位またはパーセント単位で表示します。</p> <p>構文: GetUsedAmount GetUsedPercent</p>
Partition Unused	<p>パーティション上の未使用領域のサイズをメガバイト単位またはパーセント単位で表示します。</p> <p>構文: GetUnusedAmount GetUnusedPercent</p>
Type	<p>現在のパーティションのファイルシステムが指定された種類であるかどうかを判断します。</p> <p>構文: {IsFAT IsFAT32 IsNTFS IsHPFS IsLinuxExt2 IsLinuxSwap IsExtended IsUnallocated IsUnformatted}</p>
Status	<p>現在のパーティションの状況がアクティブであるか非表示であるかを判断します。</p> <p>構文: {IsActive IsHidden}</p>
Primary/Logical	<p>現在のパーティションのクラスが基本クラスであるか論理クラスであるかを判断します。</p> <p>構文: {IsPrimary IsLogical}</p>

クエリー	使用目的と構文
Unallocated Size	<p>空き領域のサイズをメガバイト単位で表示します。</p> <p>構文: GetSelectedUnallocatedSize</p>
Unallocated Number	<p>指定された空き領域の領域番号を表示します。返された値は、Select Partition { 番号 } コマンドで使えます。戻り値0は、その空き領域が存在しないことを示します。たとえば、最後の選択の後に空き領域がない場合、GetUnallocatedNumber Next コマンドは0を返します。</p> <p>構文: GetUnallocatedNumber { <Number> First Last Largest Next Previous After Selected Partition Before Selected Partition }</p>

例

たとえば、次のコードでは、変数 dNumParts が定義されて、ディスク 1 上にあるパーティションの合計数が割り当てられます。

```
Dim dNumParts
Select Disk 1
dNumParts = GetTotalPartitions
```

次のコード例では、パーティション C: が選択されます。FAT パーティションは FAT32 に変換されます。

```
Select Partition C
If IsFAT Then
Convert to FAT32
End If
```

次の例では、ディスク 1 上の最大の空き領域が選択されています。1000MB を超える大きさの場合は、それぞれが空き領域の約半分を使う 2 つのパーティションが作成されます。

```
Select Disk 1
Select Unallocated Largest
If GetSelectedUnallocatedSize > 1000 Then
Create /FS=FAT /Size = GetSelectedUnallocatedSize / 2
Select Unallocated After Selected Partition
Create /FS=FAT
End If
```

スクリプトに関する注意事項

必須ではありませんが、変更される予定の各パーティションをそれぞれスクリプトの先頭で検査することを推奨します。スクリプトファイルはエラーが発生すると終了するため、最初にそれぞれのパーティションを検査することで、エラーが見つかる前にスクリプトによって変更されるのを防止できます。

パーティションはシリンダの境界上で始まる必要があります。たとえば、10 MB を指定した場合に実際の値は 10.2 MB になることがあります。指定した値と実際の値との差は、ディスクの種類によって変わります。

スクリプトオプションの 1 つに値を指定する場合は、その値を 1 シリンダ上回るまたは下回る誤差が許容されます。つまり、指定した値を中心に 2 シリンダの範囲の誤差が許されます。たとえば、サイズ変更で 10MB を指定したときにシリンダが 0.5MB だった場合、パーティションを 9.5MB 以上にサイズ変更できれば操作は正常に完了します。実際の範囲は、9.5MB から 10.5MB です。この範囲内でパーティションのサイズ変更ができない場合は、エラーが返されます。

通常の処理操作では、スクリプトに指定された変更内容を加えた後でコンピュータを再起動できない場合は、スクリプトはエラーで終了します。このような状態は、OS/2 を使用していて、ユーティリティディスクからコンピュータを起動したときなどに DOS.SYS ファイルが CONFIG.SYS にない場合に発生します。キーボードから手動で再起動することに問題がなければ、スクリプトの先頭スクリプトステートメントとして ALLOW MANUAL REBOOT を指定してください。

コマンドはすべて 1 行以内で指定してください。コマンド行を 2 行に分けることはできません。スクリプト行の最大長は、180 バイトです。これはすべてのコマンドに対して十分な値であるはずです。

番号でパーティションを選択するときには、十分な注意が必要です。ドライブレターまたはラベルのない空き領域やパーティションを選択する場合も、番号で選択できます。しかし、パーティションを番号で選択すると、スクリプト内で番号が変更されることがあるため問題があります。パーティション 1 と 2 の間に新しくパーティションを作成した場合、作成されたパーティションがパーティション 2 になり、元のパーティションがパーティション 3 になります。Select Unallocated After Selected Partition および Select Unallocated Before Selected Partition コマンドを使用する場合は、空き領域をパーティション番号で選択することをお奨めします。ほとんどの処理操作で、操作の完了時に選択されているパーティションがその操作の対象として選択したパーティションです。たとえば、Resize と Move では操作の対象として選択したパーティションがその操作の後にも必ず選択されたままです。Create コマンドの場合は、作成したパーティションが空き領域の先頭にあっても最後にあっても、コマンドの実行後にはそのパーティションが選択されます。操作後に選択されるパーティションに疑問があれば、DOS(緊急用ディスク)バージョンのプログラム(スクリプトなし)を使って、テストマシン上で同じ操作を実行し、操作後に選択されるパーティションを確認することができます。Show Partitions コマンドを使用して、現在のパーティションの状態を表示できます。

サンプルスクリプト

シナリオ 1: 一般的な例

基本ドライブの C:、D:、および E: があります。ディスクには空き領域がありません。D: から 10MB を取って、それを E: に追加します。

```
SCRIPT1.PQS
//最初に操作するパーティションをチェックする
Select Partition D
Check
Select Partition E
Check
//変更する最初のパーティションを選択する
Select Partition D
//そのパーティションを 10 MB だけ縮小する
Resize Smaller 10
//10 MB 追加するパーティションを選択する
Select Partition E
//パーティションを可能な限り左に移動し、作成される空き領域が右端（最後）に配置
//されるようにする
Move Left Max
//利用可能な領域を最大にする
Resize Larger Max
```

シナリオ 2: 一般的な例

ドライブに 1 つの大きな C: パーティションがあります。このドライブは 1.2 GB です。このパーティションにあるのは 300 MB のデータだけです。このパーティションを 450MB に縮小し、論理ドライブ D: と E: を作成します。E: ドライブを 300 MB にして、残りの領域を D: ドライブとします。

```
SCRIPT2.PQS
//最初にパーティションをチェックする
Select Partition C
Check
//パーティション C: は既に選択されているのでこれを 450 MB に縮小する
Resize 450
//サイズ変更後もパーティション C: が選択された状態となっているので
//C: の後に作成された空き領域を選択する必要がある
Select Unallocated After Selected Partition
```

```
// 拡張パーティションをデフォルト値のサイズで作成する。これが現在選択している
// 空き領域の全域となる
Create /FS=EXTENDED
// 現在拡張パーティションが選択されている。この拡張パーティションの中で次の空き
// 領域を選択する
Select Unallocated After Selected Partition
// 初めに必要な 300 MB のパーティションを現在選択されている空き領域の末端に作成
// する。これがパーティション E: となる
Create /FS=FAT /Label="DBFILES" /Size=300 /Position=END
// 拡張パーティション内の残りの空き領域を選択する。最後のパーティションが空き
// 領域の末端に作成されているので選択したパーティションの前の空き領域に移動する
Select Unallocated Before Selected Partition
// 残りの空き領域にパーティションを作成する
Create /FS=FAT /Label="APPS"
```

シナリオ 3: 一般的な例

1つの物理ドライブ上にパーティション C:、D:、E: があります。ディスクには空き領域がありません。

C: パーティションは基本パーティションで、D: パーティションと E: パーティションは拡張パーティションにある論理ドライブです。C: パーティションにある 40MB の空き領域に F: パーティションを作成します。F: を FAT パーティションにして「DATA」というボリュームラベルをつけます。ドライブが新しいのですべての操作に対して不良セクタのテストを省略します。

```
SCRIPT3.PQS
// 最初にすべてのパーティションをチェックする
Select Partition C
Check
Select Partition D
Check
Select Partition E
Check
// このドライブでパーティションが既に選択されているため、このドライブに対する
// 規定の不良セクタテストをオフに設定できる
Set Default Bad Sector Test State Off
// C: パーティションを選択してそれを 40 MB だけ縮小する
Select Partition C
Resize Smaller 40
```

```
// 拡張パーティションを選択して左の境界をC: パーティションの右端に最大限移動して
// 拡張パーティションの中に空き領域を確保する拡張パーティションを選択するには、
// 最初にドライブを選択してから、パーティションを選択しなければならない
Select Disk 1
Select Partition Extended
Resize Left Boundary Max
//D: パーティションを選択してそれを左に移動する
// 必ず拡張パーティションをC: パーティションにぴったりとつけてD: とE: パーティ
// ションの間に空き領域を配置する
Select Partition D
Move Left Max
//E: パーティションを選択して左に最大限移動して拡張パーティション内で空き領域が
//E: の末端に配置されるようにする
Select Partition E
Move Left Max
// 現在空き領域はE: の後にある
// ここにF: パーティション (論理ドライブ) を作成できる
//E: の後に空き領域を移動する
Select Unallocated After Selected Partition
// 規定値を使用して、DATA という名前のFAT パーティションを作成する。この場合、
// 空き領域で利用できる全サイズが使用される
Create /FS=FAT /Label="DATA"
```

シナリオ 4: 一般的な例

100 MB の基本パーティション C: があります。次のパーティションは NT_OS という名前の非表示パーティションで、100 MB の基本 FAT パーティションです。他に拡張パーティションの中に 70 MB の論理ドライブ D: と E: があります。

C: パーティションと非表示パーティションを 60 MB に縮小して、D: パーティションに 40 MB を追加し、残りの空き領域に F: パーティション (NTFS) を作成します。更に非表示パーティションを FAT から NTFS に変換します。コンピュータをプログラム制御の下で再起動できるかどうかにはこだわりません。

```
SCRIPT4.PQS
// 手動による起動が有効であることを知らせる
Allow Manual Reboot
// 最初にすべてのパーティションをチェックする
Select Partition C
Check
// 引用符に入ったボリュームラベルを使用して非表示パーティションを選択する
Select Partition "NT_OS"
```

```

Check
Select Partition D
Check
Select Partition E
Check
//C: パーティションを選択してそれを 60 MB に縮小する
Select Partition C
Resize 60
// 非表示パーティションを選択する
Select Partition "NT_OS"
// このパーティションを移動して C: パーティションにぴったりとつける。C: パーティ
// ションはサイズが変更されているため、新しく作成された空き領域は非表示パーティ
// ションの後に配置される
Move Left Max
// 非表示パーティションを 60 MB にサイズ変更する
Resize 60
// パーティションを FAT から NTFS に変換する
Convert To NTFS
// 拡張パーティションを広げて、空き領域が拡張パーティション内に存在するようにする
Select Disk 1
Select Partition Extended
Resize Left Boundary Max
//D: パーティションを移動して非表示パーティションを拡張パーティションにぴったり
// とつける
Select Partition D
Move Left Max
//40 MB を D: パーティションに追加する
Resize Larger 40
//E: パーティションを D: パーティションの次に移動する
Select Partition E
Move Left Max
// 現在空き領域を拡張パーティションの最後で利用できるため、F 論理ドライブを作成
// できる空き領域を選択します。
Select Unallocated After Selected Partition
// NTFS パーティションを作成する
Create /FS=NTFS

```

シナリオ 5: Cluster Analyzer

3.2 GB のドライブがあります。2 MB の基本パーティションと 1 GB の基本 FAT パーティション C: があります。その他のパーティションとして 1 GB の非表示基本 FAT パーティションがあります。複数のオペレーティングシステムを起動する予定で、3 番目のパーティションには別のオペレーティングシステムをインストールしています。論理ドライブを持つ拡張パーティションもあり、それがドライブ領域の残りの部分をすべて使用しています。

2 つの FAT パーティションを解析して、クラスタによる浪費を削減できるかどうかを調べます。

```
SCRIPT5A.PQS
// パーティション 2 のクラスタによる浪費を表示する
Select Disk 1
Select Partition 2
Cluster Analyzer /ShowClusterWaste
// パーティション 3 のクラスタによる浪費を表示する
Select Disk 1
Select Partition 3
Cluster Analyzer /ShowClusterWaste
```

クラスタによる浪費を削減する方法がわかったので、再度 **Cluster Analyzer** を使用して浪費を削減します。3 番目のパーティションを推奨されたサイズに設定して、2 番目のパーティションのクラスタサイズを 8K に設定します。

```
SCRIPT5B.PQS
// パーティション 3 を推奨されたクラスタサイズに設定する
Select Disk 1
Select Partition 3
Cluster Analyzer /SetToRecommended
// パーティション 2 のクラスタサイズを 8K に設定する
Select Disk 1
Select Partition 2
Cluster Analyzer /ClusterSize=8
```

シナリオ 6: Copy

新しいドライブのインストールが終了しています。ドライブ 1 上の最初の 3 つのパーティションをドライブ 2 にコピーします。ドライブ 2 はフォーマット済みで空き領域です。

```
SCRIPT6.PQS
// ディスク 1 を選択したら、次にパーティション 3、2、1 を選択、診断する
Select Disk 1
Select Partition 3
```

```

Check
Select Partition 2
Check
Select Partition 1
Check
// コピー先のディスク 2 を選択し、パーティション 1 をコピーする
Select Destination Disk 2
Select Destination Unallocated First
// 最初のパーティションをコピーする
Copy
// ディスク 1、パーティション 2 を選択する
Select Disk 1
Select Partition 2
// コピー先のディスク 2 を選択し、パーティション 2 をコピーする
Select Destination Unallocated First
// 2 番目のパーティションをコピーする
Copy
// ディスク 1、パーティション 3 を選択する
Select Disk 1
Select Partition 3
// コピー先のディスク 2 を選択し、パーティション 3 をコピーする
Select Destination Unallocated First
// 3 番目のパーティションをコピーする
Copy

```

シナリオ 7: Info

情報を取得する対象として FAT C: パーティションがあります。

パーティションの大きさや詳細については明らかではありません。ディスクの使用状況、クラスタによる浪費、パーティションの情報、およびファイルシステムの情報を必要としています。

```

SCRIPT7A.PQS
Select Disk 1
Select Partition 1
Info /Usage /Waste /Partition /FS

```

次のようなスクリプトでも同様な結果が得られます。

```

SCRIPT7B.PQS
Select Disk 1

```

```
Select Partition 1
Info /Usage
Info /Waste
Info /Partition
Info /FS
```

シナリオ 8: 基本パーティションにディスク上の未使用領域を追加する

ディスク 1 には、基本パーティションと未知数の論理パーティションがあります。このスクリプトでは、各論理パーティションから (利用できるなら) 10MB の未使用領域を切り離して、基本パーティションにその未使用領域を追加します。

```
SCRIPT8.PQS
// 最後のパーティションを選択
Select Disk 1
Select Partition Last
// ループを使って各論理パーティションから (可能なら) 10MB の空き領域を削除する
Do While IsLogical
    If GetUnusedAmount > 10 Then
        Resize Smaller 10
        Move Right Max
    End If
    Select Partition Previous
Loop
// 論理パーティションがちょうど収まるように拡張パーティションをサイズ変更する
Select Partition Extended
Resize Left Boundary Min
// 未使用領域を含むように基本パーティションを拡張
Select Partition Previous
Resize Larger Max
```