

# **VBMan Controls for Btrieve**

## **Programming manual**

**Version 3.50**



## 目次

---

目次 .....	2
<b>製品概要 .....</b>	<b>8</b>
はじめに .....	8
システムの構造について .....	8
MICROSOFT .NET対応について .....	8
対応言語について .....	9
VBMANツール・ボックス .....	9
VBMAN DDFビルダー .....	10
BTRIEVEデータ型について .....	10
データ・アクセス・メソッド/関数 .....	11
使用権 .....	11
ユーザーサポート .....	12
保証規定 .....	13
販売元 .....	14
開発元、ユーザーサポート .....	14
<b>インストール .....</b>	<b>15</b>
システム条件 .....	15
インストール手順 .....	15
依存モジュール・ランタイムについて .....	15
ライセンス認証 .....	16
インストールファイルについて .....	16
システム設定ファイル .....	17
VBMAN.INIファイル .....	17
Actian Zen対応について .....	18
<b>チュートリアル .....</b>	<b>20</b>
BTRIEVEデータベースの定義 .....	20
概要 .....	20
VBMan DDFビルダーによるレコード定義 .....	20
Visual C# .NETでデータ参照フォームを作成 .....	22
Visual Basicでデータ入力フォームを作成 .....	27
<b>サンプルプログラムについて .....</b>	<b>33</b>
.NET用サンプルについて .....	33
VISUAL BASIC 6.0用サンプルプログラムについて .....	34
ASP用サンプルプログラムについて .....	34
<b>コンパチビリティ .....</b>	<b>36</b>
VISUAL BASIC.NETへの移行 .....	37

VBMAN ACTIVEX CONTROL FOR BTRIEVE バージョン 3.Xからの移行 .....	37
VBMAN FOR BTRIEVE/OCX32 バージョン2.0からの移行 .....	37
変換ツール .....	38
手動によるフォーム変換 .....	38
VBMAN FOR BTRIEVE/VBX バージョン1.xからの移行 .....	40
フォームの変換 .....	40
データアクセス関数の移行 .....	41
VBMan DDFビルダーの機能 .....	42
<b>カスタム・コントロール・リファレンス .....</b>	<b>44</b>
データベース・コントロール .....	44
ボタン・コントロール .....	45
エディット・コントロール .....	47
スクロール・コントロール .....	48
チェック・ボックス・コントロール .....	49
オプション・ボタン・コントロール .....	50
リスト・ボックス・コントロール .....	50
コンボ・ボックス・コントロール .....	52
ピクチャー・コントロール .....	54
拡張リスト・ボックス・コントロール .....	55
拡張コンボ・ボックス・コントロール .....	57
グリッド・コントロール .....	59
<b>カスタム・プロパティ・リファレンス .....</b>	<b>62</b>
Abort .....	62
AlignFixedCells .....	62
AllOpen .....	63
AllowDelete .....	63
AllowUpdate .....	63
AllowUserResizing .....	64
AutoFirst .....	64
BlobPath .....	65
CacheDDF .....	65
CellMaxLength .....	66
ClearControlData .....	66
ConfirmMsg .....	67
DbCompareCase .....	67
DbConditions .....	68
DbDelimitChar .....	69
DbExOperation .....	70
DbField .....	70
DbIndex .....	71
DbLeaveResults .....	71
DbListFields .....	72

DbListIndex.....	72
DbListSource.....	73
DbListTable.....	73
DbLocks.....	73
DbMaxRecords.....	74
DbOperation.....	75
DbPhysicalPosition.....	75
DbResultID.....	76
DbSavePosition.....	76
DbSkipRecords.....	77
DbSource.....	77
DbTable.....	78
DbTabStops.....	78
DbUpdateOption.....	79
DbValueFalse.....	79
DbValueIndeterminate.....	80
DbValueTrue.....	80
DDFDir.....	80
DistName.....	81
Format.....	81
FormatOption.....	82
FormatString.....	82
NumericMask.....	83
OpenMode.....	83
OwnerName.....	84
ReadOnly.....	84
SetColNames.....	85
SetColWidth.....	85
SetRecordNumber.....	85
TextAlign.....	86
UpperCase.....	87
<b>カスタム・イベント・リファレンス.....</b>	<b>88</b>
AfterColUpdate.....	88
AfterDelete.....	88
AfterUpdate.....	88
BeforeColUpdate.....	89
BeforeDelete.....	89
BeforeUpdate.....	90
BtrieveError.....	90
Format.....	91
SetData.....	92
<b>カスタム・メソッド・リファレンス.....</b>	<b>94</b>
DbAbortTransaction.....	94

DbAccess .....	94
DbAccessByID.....	95
DbBeginConCurTransaction.....	97
DbBeginTransaction .....	97
DbClearControlData .....	97
DbClearFieldBuffer .....	98
DbClose.....	98
DbCloseAll .....	99
DbCreate .....	99
DbEndTransaction .....	100
DbExAccess .....	100
DbExFree .....	104
DbExGetFieldData .....	104
DbExGetPosition.....	105
DbExSetMode.....	105
DbFindPercentage.....	106
DbGetByPercentage .....	107
DbGetDataSize .....	108
DbGetDataType.....	108
DbGetDirect.....	109
DbGetFieldData .....	110
DbGetFieldDataByID .....	110
DbGetFieldName .....	111
DbGetIndexName .....	112
DbGetModulePath .....	112
DbGetNumOfField.....	113
DbGetNumOfIndex .....	113
DbGetNumOfRecords .....	114
DbGetNumOfTable .....	114
DbGetPosBlock .....	115
DbGetPosition.....	115
DbGetRecord .....	116
DbGetRecordImage.....	117
DbGetRecordLength.....	118
DbGetTableName .....	118
DbIsNull .....	119
DbIsOpen .....	119
DbLoadDDF .....	120
DbLogIn.....	120
DbLogOut.....	121
DbOpen .....	121
DbOpenAll .....	121
DbReset .....	122
DbSetFieldData .....	122
DbSetFieldDataByID.....	123

DbSetFileName .....	124
DbSetLockBias.....	125
DbSetRecord .....	126
DbSetRecordImage .....	127
DbTransferFromControl.....	128
DbTransferToControl.....	129
DbUnlock .....	129
<b>バージョン1.Xデータ・アクセス関数.....</b>	<b>131</b>
DbAbortTransaction .....	132
DbAccess .....	132
DbAttachControl .....	133
DbBeginTransaction.....	134
DbClearControlData.....	135
DbClearFieldBuffer .....	135
DbEndTransaction .....	135
DbExAccess .....	137
DbExFree .....	141
DbExGetFieldData .....	141
DbExGetPosition.....	142
DbExSetFormattedString .....	142
DbFindPercentage.....	144
DbGetByPercentage .....	145
DbGetDataSize .....	146
DbGetDataType.....	146
DbGetDirect.....	148
DbGetFieldData .....	149
DbGetFieldName .....	150
DbGetPosition.....	151
DbGetVersion.....	152
DbSetFieldData .....	153
DbSetFormattedString .....	154
DbTransferFromControl.....	155
DbTransferToControl.....	155
DbUnlock .....	156
<b>作成したアプリケーションの配布 .....</b>	<b>157</b>
<b>DDFビルダー・リファレンス.....</b>	<b>159</b>
DDFビルダーの起動 .....	159
新規DDFの作成 .....	159
フィールドの定義.....	160
インデックスの定義 .....	161
既存DDFの編集 .....	162
データ変換モードの動作について.....	163

テーブルの搬入・搬出.....	163
DBMAGICデータ辞書を搬入.....	164
VISUAL BASIC用フォームの生成.....	165
フィールドIDリストの生成.....	165
システム・オプションについて.....	166
<b>APPENDIX.....</b>	<b>169</b>
A. MICROSOFT VISUAL STUDIO.NET 制約事項.....	169
B. ACTIVE SERVER PAGESからの利用.....	169
概要・制約事項.....	169
ASPスクリプトの作成手順.....	170
C. FAQ – よくあるご質問.....	172
D. バージョン履歴.....	178
<b>エラー・メッセージ.....</b>	<b>180</b>
<b>VBMANエラー・コード.....</b>	<b>187</b>
<b>DDFビルダー・メッセージ.....</b>	<b>190</b>
<b>BTRIEVEステータス・コード.....</b>	<b>197</b>
<b>VBMAN CONTROLS FOR BTRIEVE 調査依頼フォーム.....</b>	<b>201</b>

## 製品概要

---

### はじめに

---

VBMan Controls for Btrieve ver 3.50をお買い上げくださりまことにありがとうございます。当製品はActian社のZenデータ・ベース・エンジンをアプリケーションから容易に利用できるように設計されたCOMコンポーネントです。13個のカスタム・コントロールとDDFビルダー、バージョン1.xコンパチブル・データ・アクセス関数で構成されます。当カスタムコンポーネントはVisual Basicや.NET Windows Forms標準のコントロールにBtrieveデータベース・アクセス機能を追加したコントロールです。VBManにおけるアプリケーションの開発ではDDF定義が終了後、フォームにVBManカスタム・コントロールを配置し、必要なプロパティを選択するだけで、実行可能なBtrieveアプリケーションが完成します。簡単なアプリケーションであれば、1行もコードせずにプログラムを完成することも可能です。

以下はVBMan Controls for Btrieve ver 3.50で追加された機能です。

- ① Windows10/11に対応。
- ② Actian Zen v15対応。
- ③ 64bit 版を ocx を追加。
- ④ Microsoft Visual Studio .NET 2017~2022に対応。
- ⑤ ネットワーク認証対応。

### システムの構造について

---

VBManは マイクロソフト社のCOM/ActiveX Control仕様に沿って作成されています。カスタム・コントロールとしては、グリッド・コントロール以外はWindowsの標準コントロールをサブ・クラスした形で実現しています。ランタイム関数、クラス・ライブラリはマイクロソフトVisual C++ .NET ATL version 14を使っています。

### Microsoft .NET対応について

---

Microsoft Visual StudioのWindows Formsによる開発環境のサポートしています。

Microsoft .NET frameworkではウェブアプリケーションの開発をASP.NETで行えます。ASP.NETでの開発は当製品のようなCOMコンポーネントを利用するのではなく、.NET framework 形式のマネージドライブラリを使って開発する方法をお勧めしています。弊社製品でActian Zen/Btrieveにアクセスすることの出来る製品としては Btrieve Classes for .NET が ございますので、こちらを利用した開発お勧めいたします。



## 対応言語について














---

VB6以降COM/ActiveX Controlをサポートする言語製品は市場に出回ったのですが現在はほぼ終了しています。マイクロソフトvb6, [vb.net](http://vb.net)とc#でご利用いただけるように調整及び動作確認をしています。

## VBManツール・ボックス

---

VBManをインストールすると、以下のカスタム・コントロールが言語アプリケーションのツール・ボックスに追加されます。これらのコントロールをフォーム、ダイアログ等に配置することにより、簡単にBtrieveにアクセス可能なアプリケーションを作成可能です。

	データベース	このコントロールをフォームに置くことにより、Btrieveデータベース・アクセスが可能となります。最初にフォームに設定します。
	編集テキスト	Btrieveのフィールドと文字データの交換をします。日付、時間、金額は自動フォーマット入力が可能です。
	コマンド・ボタン	Btrieveのオペレーション（追加、削除など）を実行します。
	水平スクロール	Btrieveデータベースのレコードの移動に使用します。
	垂直スクロール	Btrieveデータベースのレコードの移動に使用します。
	チェック・ボックス	論理型データを設定するチェック・ボックスです。
	オプション・ボタン	論理型、整数型のデータを設定します。
	リストボックス	Btrieveデータベースの指定フィールドのリストを表示、選択します。
	コンボ・ボックス	Btrieveデータベースの指定フィールドのリストを表示、選択します。
	ピクチャー	Btrieveデータベースの情報により、ビットマップ、メタファイル、アイコンを表示します。
	拡張リストボックス	BtrieveのExtendedオペレーションに対応したリストボックスです。
	拡張コンボボックス	BtrieveのExtendedオペレーションに対応したコンボボックスです。
	グリッド	BtrieveのExtendedオペレーションでグリッドにデータを表示します。

## VBMan DDFビルダー

---

VBMan DDFビルダーはBtrieve 6.15当時のBtrieveデータ型のみサポートする仕様になります。Actian Zen をお使いのお客様はDDFの編集、データベース入力には ZenのDDFビルダーやControl Centerをお使いください。

## Btrieveデータ型について

---

VBMan Controls for BtrieveはBtrieveの拡張データ・タイプとして定義される以下のデータ・タイプをサポートします。データ型の内部形式はActian Zen SDK Btrieve API ドキュメントを参照してください。

データ型	特記事項
String	
Integer	サイズは1,2,4,8バイトをサポート
Float	サイズは4,8バイトをサポート
Date	yyyy/mm/ddまたはyy/mm/ddフォーマット
Time	Hh:mm:ssフォーマット
Decimal	
Money	内部表現はDECIMALと同じ。VBManEditにFormat文字列を指定しない場合に金額表示形式の自動フォーマット
Logical	1バイト型をサポート
Numeric	
Numericsa	
Numericsts	
Bfloat	
Lstring	指定するサイズは先頭1バイトの長さを含む
Zstring	指定するサイズは終端のヌル文字を含む
Unsigned Binary	
Auto Increment	
Note	3.4000バイトまで。デリミタはヌルをサポート
LVar	3.4000バイトまで
Bit	

Currency	
Blob	Long varcharに対応。

## データ・アクセス・メソッド/関数

---

アプリケーション作成において、たとえば集計処理などを記述する際には、VBMan for Btrieve/VBX ver 1.xではDDFにしたがって、容易にBtrieveにアクセス可能とする、データ・アクセス関数が用意されていました。VBMan Controls for Btrieveでは、これらの関数はデータ・ベース・コントロールのメソッドとして提供されます。またバージョン1.5までは関数のパラメータはすべて整数のIDでしたが、32Bit環境で文字列領域に余裕があると判断し、これらのパラメータはすべて文字列で指定可能となりました。以下はVBMan Controls for BtrieveをVisual Basicから利用する際のコード例です。

```
Dim Code$, rc As Integer
Code$ = Text1.Text
With VBMBtr1
    Call .DbSetFieldData("商品","コード", Code$)
    rc = .DbAccess( BTR_GET_EQUAL, "商品", "コード")
    If rc <> 0 Then
        MsgBox "Btrieveエラー " & CStr(rc)
        Stop
    End If
End With
```

## 使用権

---

使用権とは、お客様が1台のパーソナル・コンピュータ・システムでVBMan Controls for Btrieveの開発環境を利用することが出来る権利です。

- VBMan Controls for Btrieveの使用権はいかなる方法によっても第三者に譲渡および貸与することは出来ません。
- 使用権はVBMan Controls for Btrieveのライセンス認証を実行したときに発効します。一度開梱した商品の返品には応じることはできません。
- ランタイム・モジュールのライセンス料は無料です。お客様のアプリケーションと一緒に配布可能なファイルは当マニュアルの「インストール」にあるモジュール一覧をご覧ください。
- 当製品の利用によるお客様の損失等に関してましては弊社および、販社システム・ラボは一切責任を負いませんのでご了承ください。

使用权は以下のいずれかの事由が起こった場合に消滅します。

1. 当ソフトウェアの不正使用により弊社に損害を与える場合。
2. 購入者が使用規定に違反した場合。
3. ライセンスキーの第三者利用。
4. 当製品のリバース・エンジニアリングの対象とした場合。

## ユーザーサポート

---

- ユーザー登録

当製品には、ユーザー登録用紙を添付しています。お買い上げのあと、できるだけ早い機会に、必要事項をご記入の上、販売会社システムラボまでファックスにてご返送いただくか、システムラボのWebSiteにてご登録をお願いいたします。(http://www.systemlab.co.jp/vbman/productvb7.htm)ユーザー登録が行われていないと、ユーザーサポートが受けられない場合がございます。

- 質問のご担当について

基本的にユーザー登録された方以外のユーザー・サポートへのお問い合わせにはご返答することは出来ません。管理者の方のお名前でユーザー登録されている場合には開発者の方1名様のみサポートの対象になります。1ユーザー版をご利用のお客場で複数のお名前の連名でのご質問をされる場合がありますが、そのような場合は不正コピー使用とみなし、サポートを打ち切らせていただく場合もございます。あらかじめご了承ください。

- お問い合わせの方法

**当製品につきましては、製品の性格上複雑なやりとりになることが多く、記録を残すためにも、電話によるユーザーサポートはいたしておりません。ファックスまたは電子メールでのお問い合わせのみ承りますのであらかじめご了承ください。**

ファックスでのお問い合わせの場合は巻末にあります「調査依頼書」に必要事項をご記入の上、テクナレッジまでファックスで送信してください。電子メールでのお問い合わせにつきましては、同様に巻末の「調査依頼書」にあります項目を記入の上、メール・アドレスは [support@techknowledge.co.jp](mailto:support@techknowledge.co.jp)宛てお送りください。

- サポート期間について

無償サポートはユーザー登録終了後、最初のサポートを開始した日から**90日間、3インシデント**を上限といたします。ただし、キャンペーン期間にお買い求めになったパッケージは追加パッケージをお買い求めになった場合はサポート期間が別途設定される場合もあります。無償サポート終了後には有償サポート・サービスもご用意しております。詳細はお手数ですが販社システム・ラボまでおたずねください。

- サポート内容について

VBManのサポートでは弊社製品に関するお問い合わせに限定させていただきます。Actian Zen/Pervasive.SQL/Btrieveの使い方に関するお問い合わせは該当サポートを提供している会社宛てをお願いいたします。また、お客様が作成されるアプリケーションの仕様に関する事

や、一般的なプログラム・ロジック等をご質問されてもユーザー・サポートではお答えすることはできません。

- 製品サポート情報の公開  
テクナレッジでは当製品の最新サポート情報を弊社webサーバーにて公開しています。  
(<http://www.techknowledge.co.jp>)また、最新バージョンのモジュールがダウンロード可能になっております。サポートにお問い合わせになる前にぜひご参照ください。
- サポートのパフォーマンスについて  
簡単なお問い合わせであれば、1労働日以内を目標にサポートをしています。調査を必要とするお問い合わせには、回答に時間がかかる場合がありますので、あらかじめご了承をお願いいたします。サポートには優先順位はありませんが弊社内に到着順、情報の蓄積があるものから処理していきます。サポート用紙に「納期が近いので」とか「回答を早く」などの記述をされた場合でも受付順が変更されることはありませんのでご了承ください。また**回答スケジュール、調査スケジュール、返答期限等の具体的な日程は当サポートでは提示出来ませんのでご了承ください。**
- 登録内容の変更について  
転居などによるご住所や電話番号など登録内容に変更が生じた場合には、郵送、ファックス、または電子メールにて、販売会社システム・ラボまでご連絡をいただきますようお願いいたします。なお、電話による口頭での連絡変更は受けかねますので、よろしく願いいたします。
- 併用される他社製品について  
当社製品と併用される、他社製品の使い方等についてのご質問をお受けすることがあります。しかし、他社製品に関しましては、お答えできない場合があります。他社製品につきましては、概当開発、販売会社または製品サポート会社にご連絡ください。

## **保証規定**

---

当製品、および付随する著作物に対して商品性及び特定の目的への適合性などについての保証を含むいかなる保証もそれを明記するしないに関わらず提供されることはありません。

当製品の著作者及び、製造、配布に関わるいかなる者も、当ソフトウェアの不具合によって発生する損害に対する責任は、それが直接的であるか間接的であるか、必然的であるか偶発的であるかに関わらず、負わないものとしします。それは、その損害の可能性について、開発会社に事前に知らされていた場合でも同様です。

# Systemlab®

SOFTWARE PRODUCTS

(株) システムラボ

東京都北区田端6-1-1 田端アスカタワー 1 2階

電話： 03-5809-0893

E-Mail： [Info@systemlab.co.jp](mailto:Info@systemlab.co.jp)

Web： [www.systemlab.co.jp](http://www.systemlab.co.jp)

---

開発元、ユーザーサポート

---

# TechKnowledge

(株) テクナレッジ

東京都世田谷区駒沢2丁目16番1号 サンドービル 9F

電話: 03-3421-7621

FAX: 03-3421-6691

E-Mail: [Info@techknowledge.co.jp](mailto:Info@techknowledge.co.jp)

Web: [www.techknowledge.co.jp](http://www.techknowledge.co.jp)

## 商標登録

本マニュアルに記載される商標、登録商標は該当会社の商標または登録商標です。

# インストール

---

VBMan Controls for Btrieve 3.5のインストール方法や条件について説明します。

## システム条件

---

VBMan Controls for Btrieveを動作させるには、以下の前提となるソフトウェア環境が必要となります。

- ① Microsoft Windows ~11 または Windows Server ~2022
- ② COM/ActiveX Controlをサポートする言語
- ③ Actian Zen または Pervasive SQL

また、アプリケーション開発においてはBtrieveについての詳しいプログラミング情報が必要になる場合が想定されます。そのような場合は Actian Zen SDK Btrieve API ドキュメントやサンプル・プログラムをご参照ください。弊社ユーザー・サポートにBtrieve自体の使用法、プログラミングの詳細、マニュアルの内容等をご質問されても返答いたしかねます。Btrieve APIやActian Zenに関する質問はエージーテックサポートをご利用ください。

## インストール手順

---

- ① Actian Zen/Pervasive SQLなどをインストールしてください。
- ② vbman350.zipの圧縮を解除します。
- ③ setup.exeを **管理者権限**で起動します。
- ④ インストーラーの質問に答えてOKボタンをクリックするとインストール終了します。
- ⑤ インストールが正常に終了するとスタートメニューに当製品のメニューが作成されます。
- ⑥ readme.htmlファイルには最新情報があります。ご一読ください。

## 依存モジュール・ランタイムについて

---

当製品の依存するランタイムは以下からダウンロードできます。Windows Serverなど一部のサーバーでネットワークから依存モジュールをインストールできない場合は以下からダウンロードしてインストールしてください。ライセンス認証プログラムのみ.NET frameworkに依存です。

Visual C++ 再頒布可能パッケージ

[https://aka.ms/vs/17/release/vc\\_redist.x86.exe](https://aka.ms/vs/17/release/vc_redist.x86.exe)

.NET framework 4.8

## ライセンス認証

---

当バージョンからネットワークライセンス認証方式になり従来のセンチネルは廃止しました。インターネットに接続したPCでメニューから「ライセンス認証」を選択して以下の画面でシステムラボからのライセンス番号を入力し認証ボタンを押してください。

## インストールファイルについて

---

VBManの導入ディレクトリを<instdir>とした場合に導入されるファイルの一覧を以下に示します。デフォルトインストールでは<instdir>は c:\program files\TechKnowledge\VBMan Controls for Btrieve 3.5となります。

お客様の作成したアプリケーションに添付して配布するモジュールには「再配布」のコラムに「可」と記述されるものに限定されます。それ以外のモジュールを配布した場合、著作権法違反となりますので十分ご注意ください。

デフォルト・パスとファイル名	内訳	再配布
<instdir>\bin\vbmbt350.ocx	32bit 版 ocx	可
<instdir>\bin\ddfio350.dll	32bit 版 DDFサポートt DLL	可
<instdir>\bin64\vbmbt350.ocx	64bit 版 ocx	可



<instdir>\bin64\ddfio350.dll	64bit 版 DDFサポート DLL	
<instdir>\bin\msflxgrd.ocx	マイクロソフトのグリッド・コントロール。DDFビルダーが使用。	可
<instdir>\man\readme.txt	VBMan最新情報などの記述	不可
<instdir>\bin\vbmdmf.exe	DDFビルダー	不可
<instdir>\bin\vbconv.exe	バージョンコンバーター	不可
<instdir>\man\vbmdmf32.hlp	VBMan DDF ビルダーヘルプファイル	不可
<instdir>\samples\vb\vbmanv1.bas	VBManバージョン1.0コンパチブル関数定義Visual Basicソース	不可
<instdir>\samples\vbmdmf32.exe	VBMan DDF ビルダー実行ファイル	不可
<instdir>\samples\regsvr32.exe	マイクロソフトのOLEカスタム・コントロール登録プログラム。	可
<instdir>\samples\vb\**	VB用サンプル・プログラム	不可
<instdir>\samples\asp\**	ASP用サンプル・プログラム	不可

## システム設定ファイル

### VBMAN.INIファイル

VBMAN.INIファイルは通常Windowsディレクトリに配置され、VBManのシステム設定に関する情報が書きこまれます。インストール直後にはこのファイルは存在しませんが、VBMan DDFビルダーの設定を書きこむ際に生成されます。書込権限のため管理者権限で実行が必要です。

VBMan for Btrieve/VBX Version 1.xではnetworkやdbpathの設定が必須でしたが、データベース・コントロールのプロパティに設定または自動判定するように変更になりましたのでご注意ください。またセクション名は[vbman350]に変更になりました。以下は設定項目とその説明です。

設定項目	詳細
float_size	Float, Bfloat型のフィールドを文字列に展開する際の浮動小数点の桁数。この値の設定が小さい場合、4バイトのFloat型では桁落ちが発生する場合がありますので適切なサイズを設定してください。デフォルトは10桁です。最大は20桁です。変換の精度はマイクロソフトのコンパイラのランタイム・ライブラリに依存しています。

sShortDate	Date型のフィールドを文字列に展開する際または文字列からDate型に変換する際の形式を設定します。yyyy/mm/ddまたはyy/mm/ddが設定可能です。デフォルトはyyyy/mm/dd形式です。
grid_max_col	VBManグリッドが扱えるカラムの最大値を設定します。デフォルトは64カラムです。初期化時に割り振るメモリの量はgrid_max_col*4 + grid_max_row*4 + grid_max_col*grid_max_row*4バイトです。32bit環境のためC/C++のランタイム・メモリの容量が上限になります。
grid_max_row	VBManグリッドが扱えるローの最大値を設定します。デフォルトは1024ローです。
sign_nibble	DECIMAL型、MONEY型の正数値を表すニブル値を設定します。設定可能な値は12または15です。デフォルト値は15となります。
Year_2000	日付け型をYY/MM/DDの形式で扱う場合、この値に1をセットすると年の値に2000を加算します。デフォルト値は0でこの場合は従来バージョンと同じく1900を加算します。このフラグはデータの登録時に参照されます。Btrieveデータ自体は2000年以上の値を保持することが可能で、VBManではYY/MM/DD表示形式の場合は2000年は00と表示されます。

上記以外にもVBMAN.INIファイルには弊社VBMan製品の設定<sup>1</sup>が書き込まれます。

以下はVBMAN.INIファイルのサンプルです。

```
[vbman350]
float_size = 15
sShortDate = yyyy/mm/dd
grid_max_col = 30
grid_max_row = 400
```

## Action Zen対応について

Action ZenではBtrieveの拡張データタイプ以外のカラムを定義することができますがVBManではそのようなデータ型のカラムはサポートされません。(longvarbinary等)

ver 3.4から「真のヌル値」(true nullable)をサポートするテーブルを扱えるようになりました。フィールドの値のヌルチェックにはDbIsNullメソッドを利用します。

Action Zen リリースレベルにより仕様詳細が変わりますのでリリースノート(readme.html)や開

<sup>1</sup> VBMan for ODBC、VBMan DDFビルダー等。

発会社のWebサイトなどで随時情報を提供しますので、ご参照ください。

# チュートリアル

---

ここでは、VBMan Controls for Btrieveを使ったフォームの作成方法について説明します。

## Btrieveデータベースの定義

---

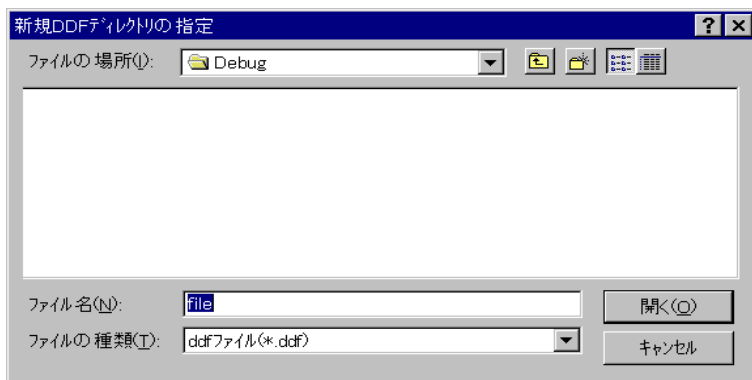
### 概要

ここではVBMan DDFビルダーを使用したデータベースのフィールドの定義方法について説明します。フォームのデザインに入る前にデータベースフィールドの定義とBtrieveデータを生成します。

### VBMan DDFビルダーによるレコード定義

VBMan for ActiveX ControlのメニューにはVBMan DDFビルダーが登録されます。DDFビルダーを実行するにはこのメニューを選択します。以下は新規にBtrieveレコードを定義する手順です。

- ① VBMan DDFビルダーを実行します。
- ② 初期メニューの「ファイル」から「DDFの作成」を選択します。
- ③ 以下のダイアログが表示されます。DDFを保存するディレクトリを指定します。
- ④ DDFディレクトリを指定するとテーブル情報の入力ダイアログが表示されます。この例では

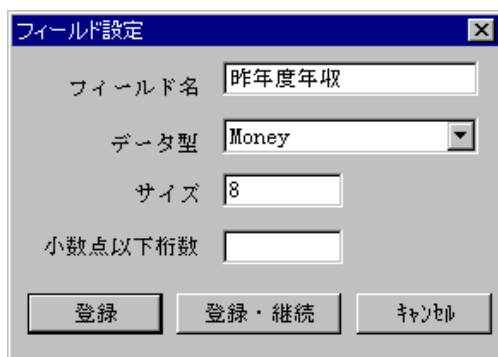


テーブル名もBtrieveファイル名もtest1と設定します。

- ⑤ 空のテーブル定義画面が表示されます。
- ⑥ 「フィールド」メニューから「フィールドの追加」を選択します。以下のようなフィールド



定義画面が表示されます。



フィールド名を入力し、データ・タイプをコンボ・ボックスから選択し、必要に応じて、データ長を入力します。文字列などの場合はデータ入力が必要ですが、たとえば、時間、日付けなどは内部的なデータのサイズは規定されているので設定の必要はありません。この場合、データ・サイズの入力カラムにはフォーカスが移動しません。小数点以下桁数についても、MONEY,DECIMAL,NUMERIC以外ではフォーカスが移動しません。

- ⑦ 上の操作を繰り返して、フィールドをすべて登録します。
- ⑧ インデックスを設定します。VBManでコントロールを使ってデータを入力する再には1個以上のインデックスの定義が必須になります。「インデックス」メニューから「インデックスの追加」を選択すると以下のようなダイアログが表示されます。



- ⑨ インデックスを設定するフィールドを選択してソート・オーダーを選択します。今回はチュートリアルなので、上記のダイアログのようにキー・タイプを設定します。
- ⑩ 「ファイル」メニューから「保存」を選択していまのフィールド定義を保存します。
- ⑪ 「オプション」メニューから「Btrieveファイル生成」を選択します。Test1.btrというファイルをリスト・ボックスから選択して「OK」ボタンを押すとBtrieveファイルtest1.btrが生成されます。
- ⑫ 以上でDDFの定義とBtrieveファイルの生成は終了です。DDFビルダーを終了させます。「ファイル」メニューから「終了」を選択します。

複数のテーブルが必要な場合はファイル・メニューから新規テーブルを選択して、同様の操作を繰り返します。

VBMan DDFビルダーの操作の詳細についてはオンライン・ヘルプをご参照ください。

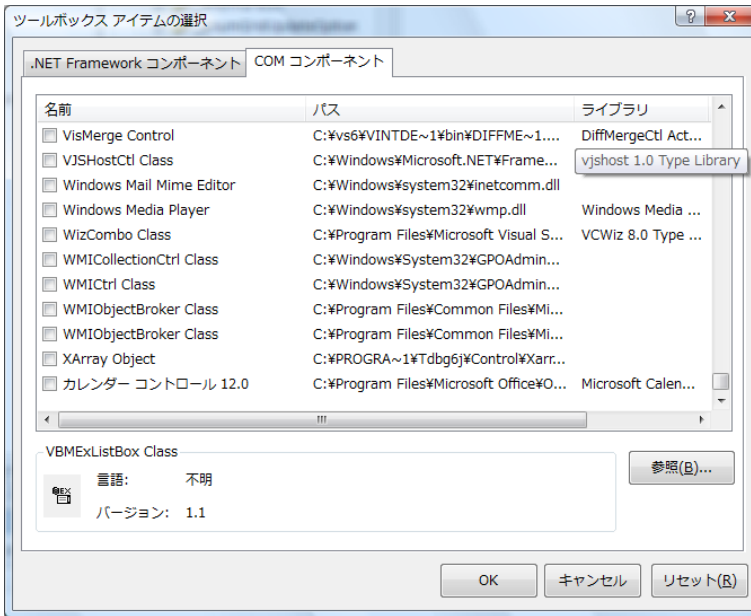
#### Visual C# .NETでデータ参照フォームを作成

ここではMicrosoft Visual C# .NETでVBMan Controls for Btrieveを使ったデータ表示フォームの作成方法を説明します。この例ではActian Zenのサンプルデータdemodataを使っています。Visual Studio 2022からIDEが64bitになりましたのでvbmbt350.ocxも64bit版をご利用ください。ビルドアーキテクチャはAnyCPUではなくx86またはx64を指定して実行してください。(0x80040154 (REGDB\_E\_CLASSNOTREG)が発生します)

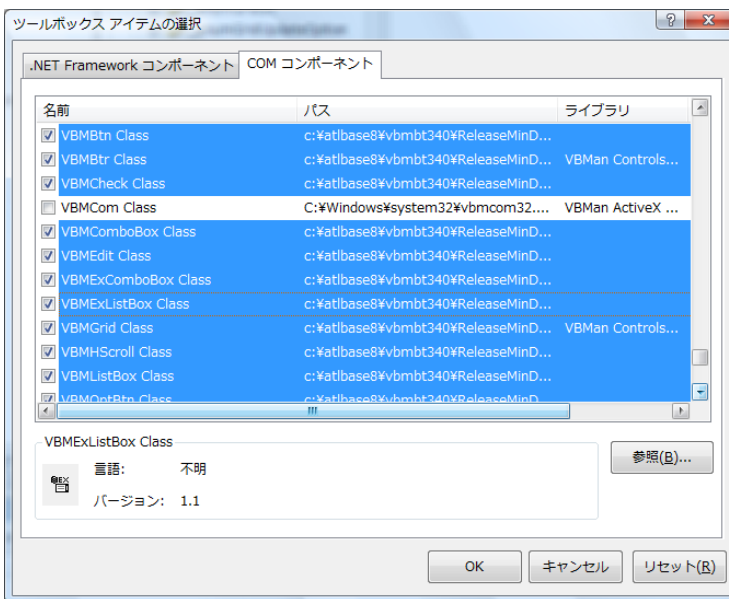
- 1) Visual Studio.NETを起動します。
- 2) 新規プロジェクトからVisual C#プロジェクトを選択しWindows アプリケーションを選

択してプロジェクトを開始します。

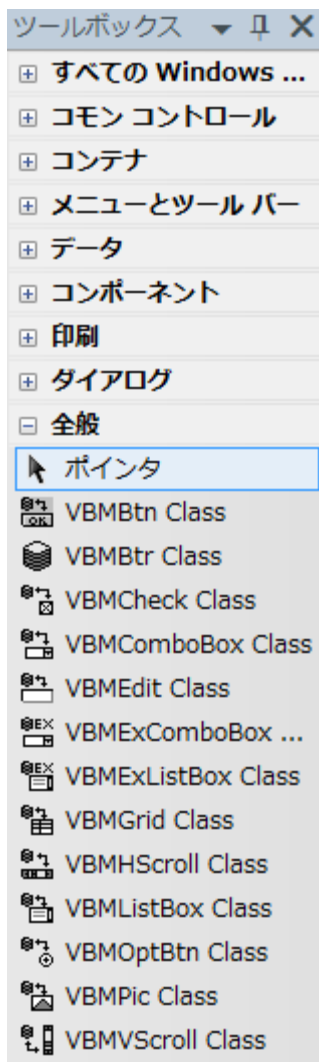
- 3) ツール・ボックスをマウス右クリックし、「ツールボックスのカスタマイズ」を選択します。
- 4) COMコンポーネントのタブで「参照」ボタンを押します。



- 5) インストールフォルダのbinにあるvbmbt350.ocxファイルを指定します。するとCOMコンポーネントのタブは以下ようになります。Visual Studio 2022をお使いの場合は bin64フォルダーのvbmbt350.ocxをご利用ください。

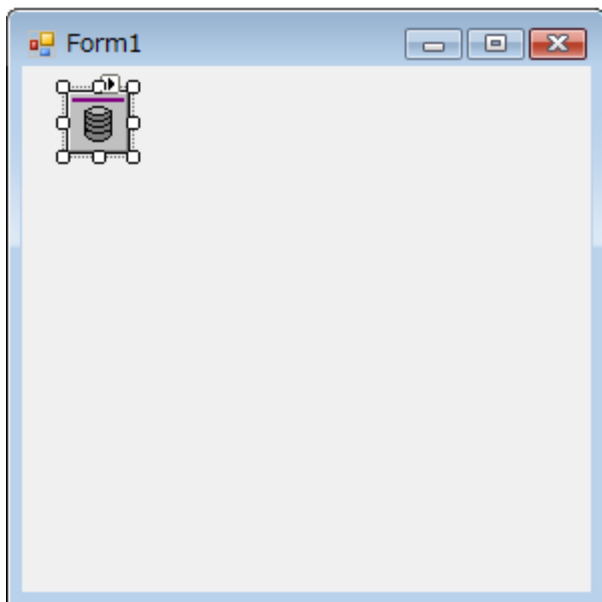


6) ツールボックスにVBManコントロールが以下のように表示されます。

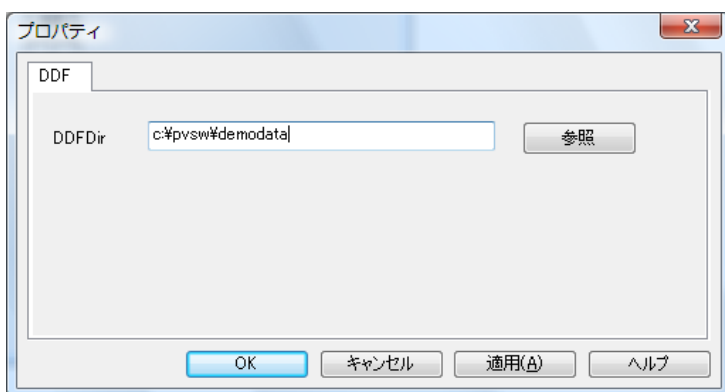


7) データベース・コントロール(VBMBtr)をWindows Formにドラッグします。

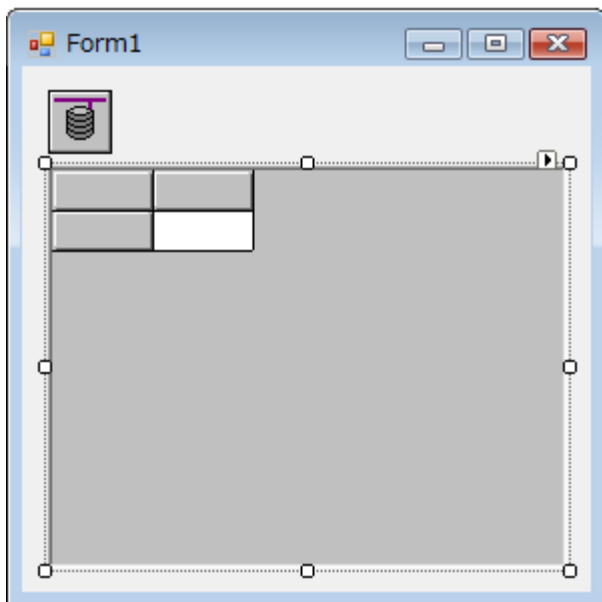




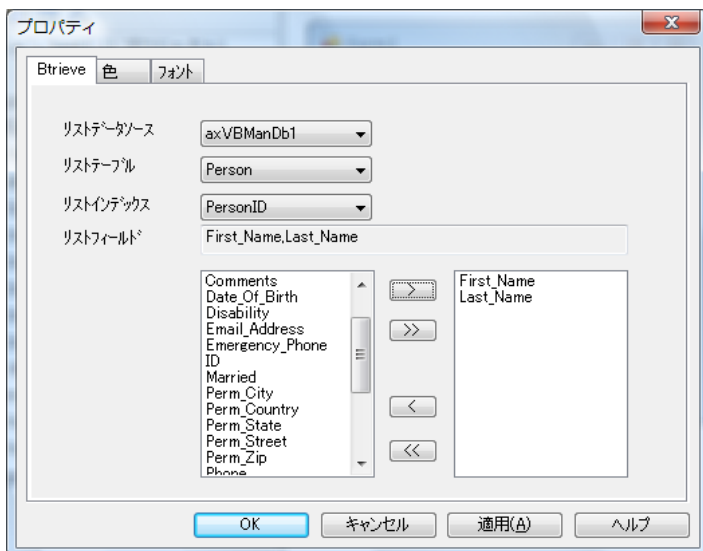
- 8) プロパティページをマウス右クリックで開き、DDFディレクトリにDEMOMDATAディレクトリを指定します。以下の画面の例はActian Zen インストール時での設定です。



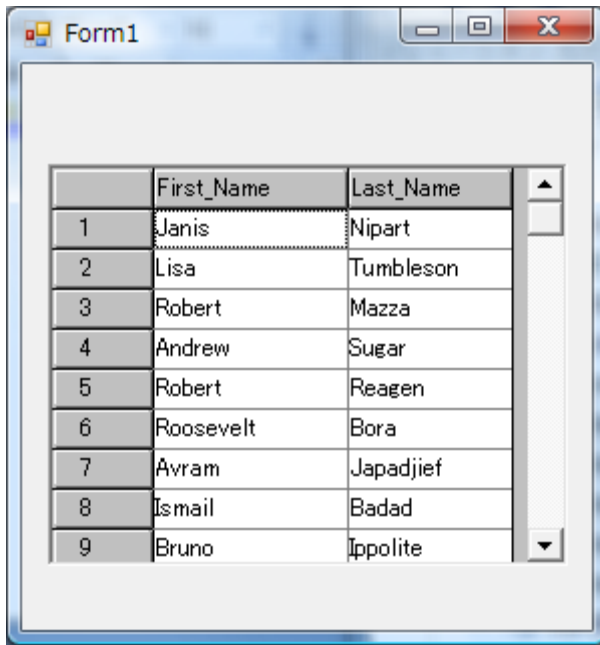
- 9) Widnows formにVBManグリッドをドラッグします。



- 10) VBManグリッドをマウスの右クリックでプロパティダイアログを開き、以下のように PersonデータベースからFirst\_name,Last\_nameカラムを選択します。インデックスにはPersonIDを指定します。



- 11) 実行します。グリッドにデータが表示されることを確認します。



12) この例ではVisual C#でプロジェクトを新規開始しましたが、Visual Basic.NETでもプロジェクト選択以外は全く同一の手順となります。

#### Visual Basicでデータ入力フォームを作成

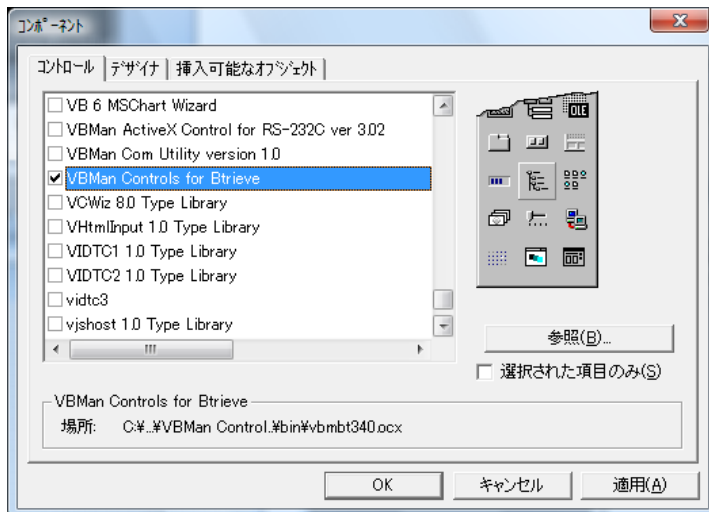
この章ではVisual Basic 6.0でVBMan Controls for Btrieveを使ったデータ入力フォームの作成方法を説明します。

- 1) Visual Basic 6.0を**管理者権限**で実行します。(Visual Basic 6がレジストリアクセスするので最近のWindowsでは管理者権限が必要です)
2. 「プロジェクト」メニューの下のカスタム・コントロールを選択します。以下のようなダイアログが表示されますので、リストから「VBMan Controls for Btrieve」を選択します。

<sup>2</sup>

リストボックスに「VBMan Controls for Btrieve 3.5」が表示されない場合は、インストールが正しく行われていない可能性があります。管理者権限のコマンドプロンプトでREGSVR32.EXEを実行してVMBMT350.OCXを登録してください。

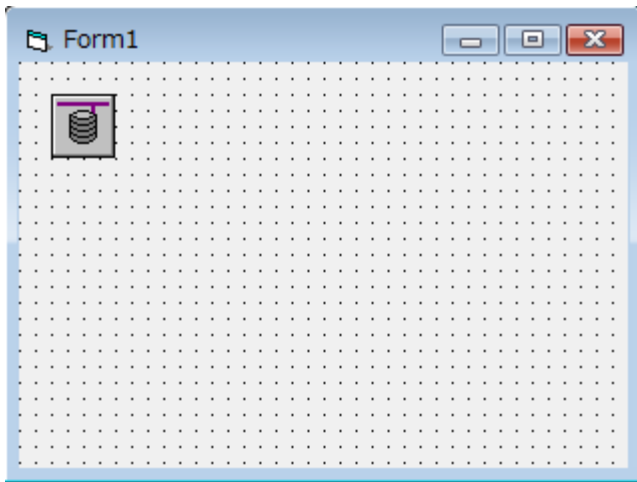
チェック・ボックスが選択されている状態であることを確認してOKボタンを押します。



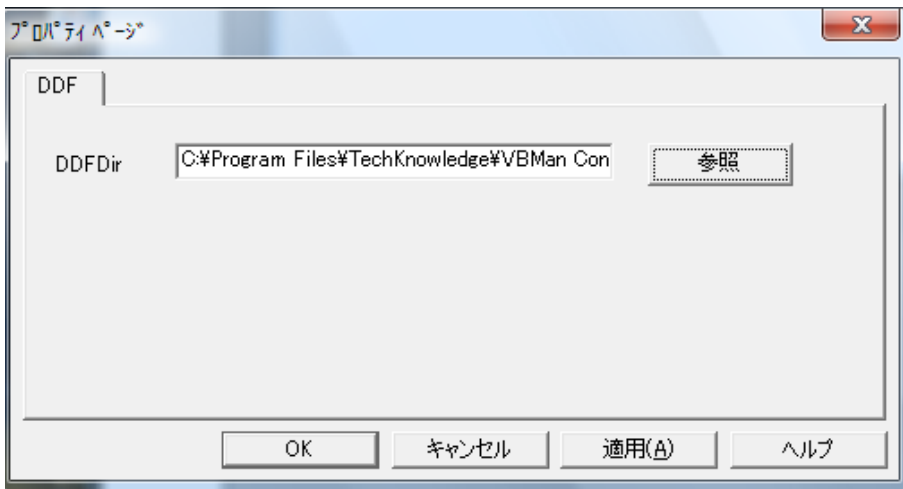
3. Visual Basicのツール・パレットにVBMan Controls for Btrieveのカスタム・コントロールのアイコンが以下のように表示されます。



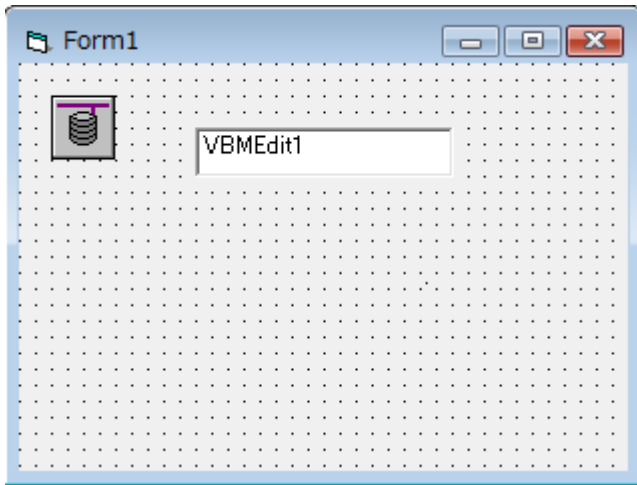
4. VBManデータアクセスコントロール（ディスクが1個だけのアイコン）をフォームにおきます。必ずこのコントロールを他のすべてのVBManコントロールより先にフォームに設定してください。このコントロールは実行時には表示されないため、フォームのデザイン時に邪魔にならない適当な位置に配置します。



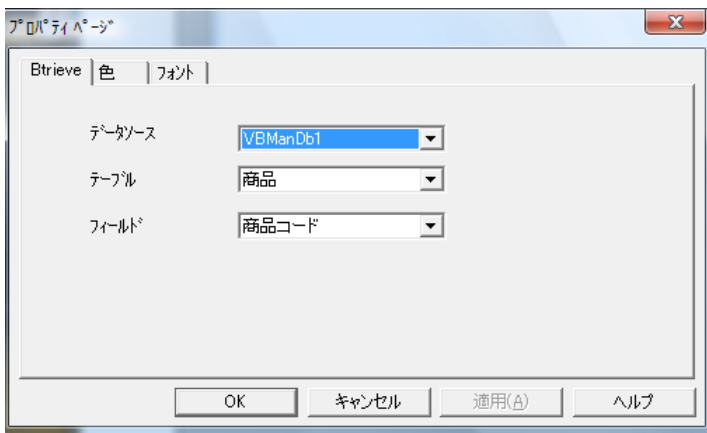
5. データベース・コントロールのDDFDirプロパティを設定します。このプロパティはドライブまで含めたDDFファイルが存在するパスを指定します。FILE.DDF, FIELD.DDF, INDEX.DDFの3つのファイルが存在するパスであることが必須となります。このプロパティが正しく設定されると、テーブル、フィールド、インデックスに関する情報をDDFからコントロールに読み込みます。これらの情報により、フォームのデザイン時にデータ・ベースに関連する情報を正確に効率よく設定することが可能になります。



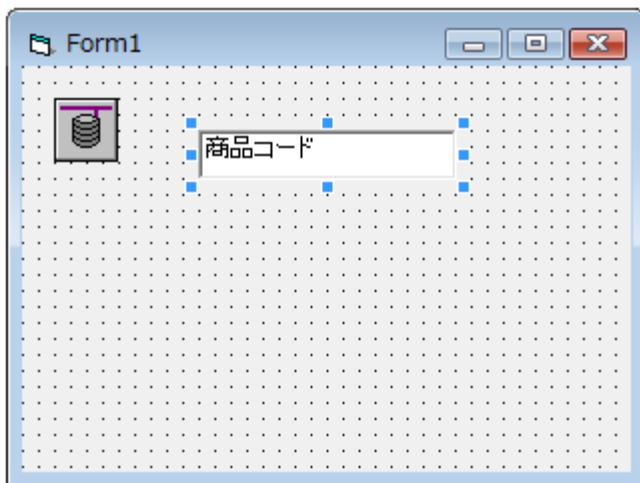
6. データの入力フィールドをフォームにおき、プロパティを設定します。この例ではエディット・コントロールをフォームに設定します。プロパティDbSource, DbTable, DbFieldの設定は必須となります。これ以降の例ではDbSourceプロパティはフォームに唯一設定しているVBManデータベース・コントロールのオブジェクト名の値を設定します。



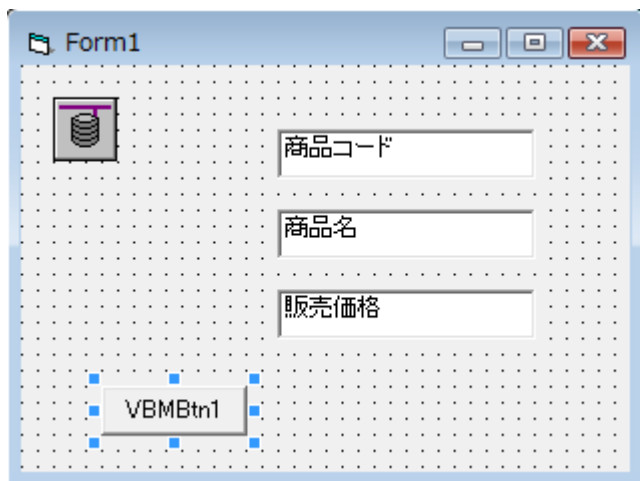
7. 以下はエディット・コントロールのプロパティ設定ダイアログです。エディット・コントロールをマウスの右ボタンでクリックして一番下のメニュー「プロパティ(P)」を選択すると表示されます。このダイアログでプロパティを設定する場合は、データソース、テーブル、フィールドの順にコンボ・ボックスから選択します。OKボタンを押すと初回のみフィールド名がエディット・コントロールのTextプロパティ値に設定されます。



8. エディット・コントロールのDbSource,DbTable,DbFieldプロパティを設定した直後のフォームは以下ようになります。



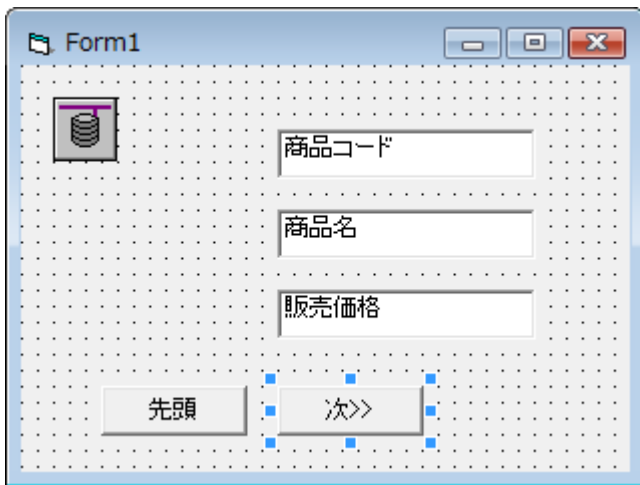
9. 該当テーブルの入力項目をフォームに設定したら、Btrieveデータベースの操作をするVBManボタンコントロールをフォームにおきます。



10. VBManボタン・コントロールでは、DbSource,DbTable,DbIndexプロパティとDbOperationの設定が必須です。プロパティ設定のダイアログではBtrieveのBtrieveオペレーションをDbOperationプロパティに設定することも可能です。以下の例では先頭レコードに移動するGetFirstオペレーションを選択しています。



11. 以下はボタンのキャプションを「先頭」と設定し、次レコードに移動するためにGetNextボタンを追加したフォームを実行している様子です。



12. 以上のように、VBManエディット、VBManボタンを複数配置することにより、簡単なフォームであれば、すぐにデータを登録したり、検索する実行可能なフォームができあがります。この時点で設定必須なプロパティが指定されていれば、実行可能です。また、いまままで説明したフォームはVBMan DDFビルダーの自動フォーム生成機能で作成することもできます。



# サンプルプログラムについて

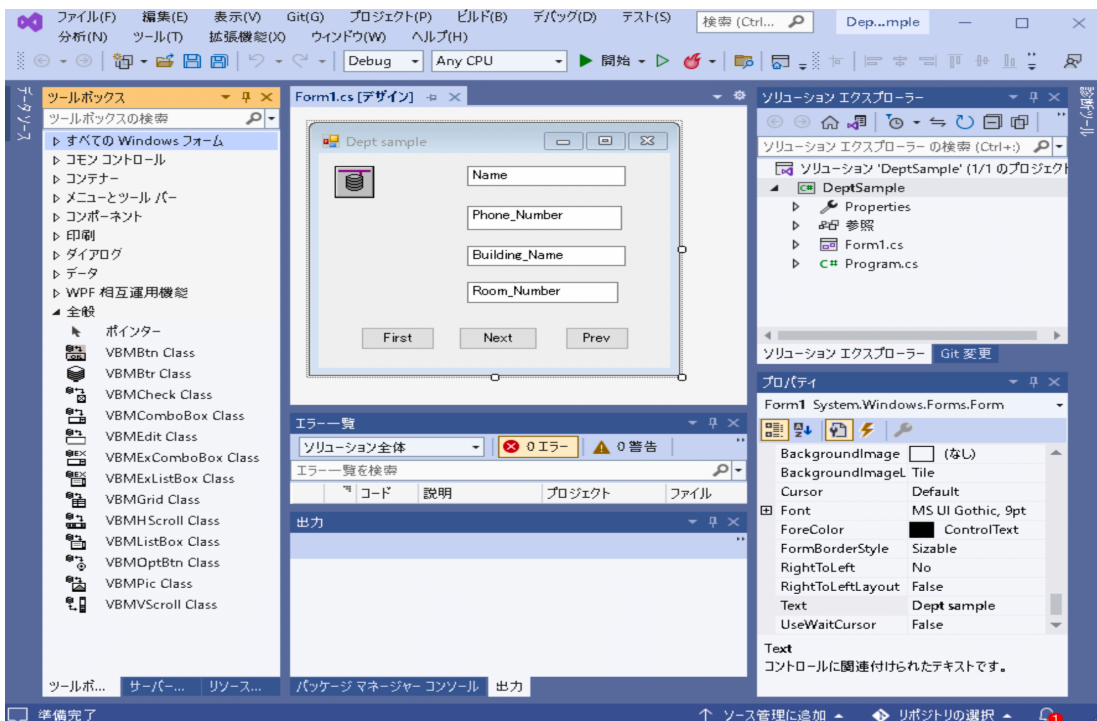
## .NET用サンプルについて

.NET 用のサンプルはVB.NETおよびC#言語用が用意されています。それぞれメニューからプロジェクトソリューションファイルへのショートカットが張られていますので、Visual Studio.NET IDEがインストールされている環境をあらかじめご用意いただければ、メニューからサンプルプログラムをごらんいただくことができます。

プロジェクトを開く前に、Visual Studio.NETのツールボックスにVBMan Controls for Btrieveのコンポーネントを登録しておく必要があります。（登録していない場合は、コンポーネントが参照できない、というエラーになります） ツールボックスへのコンポーネントの登録方法は、前章の「Visual C# .NETでデータ参照フォームを作成」をご参照ください。

サンプルプログラムビルド時の注意点ですが参照しているラッパーコンポーネント(AxVMBMtrLib.DLL)の「ローカルコピー」プロパティにFalseが設定される場合にはTrueに設定していただくとエラーなく、簡単にサンプルプログラムをコンパイルして実行できます。

以下はVisual Studio 2022でのスクリーンショットになります。



## Visual Basic 6.0用サンプルプログラムについて

---

Visual Basic 6.0用のサンプルはインストールディレクトリ（デフォルトでは\Program Files\TechKnowledge\VBMan Controls for Btrieve 3.5）以下のsamples\vbフォルダーにあります。6種類のサンプルプロジェクトが収録されています。（拡張子VBPファイル）各サンプルはDDFディレクトリとしてデフォルトインストールの場合のフォルダーに固定されていますので、デフォルトディレクトリ以外のインストールでは、DDFディレクトリを該当ディレクトリに変更してください。

## ASP用サンプルプログラムについて

---

当製品のサンプルには以下のようなファイルが添付されています。以下はサンプルの簡単な説明です。

Sample1.asp	Btrieveデータを<table>タグで表示するサンプルです。ブラウザからはsample1.aspファイルを直接指定します。お客様の環境で動作させるにはDDFDirプロパティの設定変更が必要です。
Sample2.htm Sample2.asp	Btrieveデータを新規登録するサンプルです。ブラウザからはsample2.htmファイルを指定します。データが入力されて「登録」ボタンをクリックするとsample2.aspがデータをBtrieveデータ・ベースに登録します。Sample2.aspのDDFDirプロパティはお客様の環境に合わせて設定変更が必要です。

サンプルを動作させるためにはBtrieveが利用可能なIISがインストールされているWindowsサーバーが必要となります。

VBMan Controls for BtrieveのASPサンプルディレクトリにある以下のファイルをサーバーのディレクトリにコピーして、そのパスをsample1.aspとsample2.aspファイルのDDFDirプロパティ設定の行に設定する必要があります。以下のファイルはVBManのインストール・ディレクトリ以下のsamples\aspディレクトリにインストールされます。

File.ddf  
Field.ddf  
Index.ddf  
Shiire.btr  
Item.btr

たとえば、上記ファイルをサーバー名 [\\jackson](#)の\ddfディレクトリにコピーした場合は sample1.asp/sample2.aspファイルのDDFDirプロパティ設定行は以下のような記述になります。

```
<% VBMan.DDFDir="\\jackson\ddf" %>
```

## コンパチビリティ

この章ではVBMan Controls for Btrieveの以前のバージョンからのVisual Basicアプリケーションの移行方法を示します。以下に各バージョンの説明をいたします。

VBMan for Btrieve/VBX スタンダード版	1994年5月に発売されました。Windows 3.1+VB2で動作するVBXです。最終バージョンは1.15となっています。
VBMan for Btrieve/VBX プロフェッショナル版	1994年11月に発売されました。Windows 3.1+VB2で動作するVBXです。スタンダード版に対してBtrieve拡張オペレーションなどが利用できる仕様に拡張されています。最終版はVB4 16bit版でも動作するように修正されたバージョン1.58となります。
VBMan for Btrieve/OCX32	1996年3月に発売されました。32bit OCXとなり、VB4 32bit版で動作するものが開発されました。対応するBtrieveのバージョンも32bit版の6.15となりました。その後VB5にも対応し、最終版は98年7月現在2.001となっています。VBXではデータ・アクセス関数でアプリケーションを作成することが多かったのでコンパチブル関数としてVBX版のソースをそのまま使える関数が用意されています。
VBMan ActiveX Control for Btrieve ver 3.0X	1998年に発売されました。ATLというクラスライブラルを採用したために、製品のソースは大幅に書き換えが必要でした。ATLを採用したのでいわゆるDLL HELL問題を解決できました。マイナーバージョン・アップでWindows2000に対応しました。Pervasive.SQL 2000のService Pack1からBtrieve APIの仕様変更がありこれにもマイナーバージョン・アップで対応しました。初期バージョンはATL ver 2.0/2.1を使ってリリースしました。後半のバージョンはATL 3.0でビルドしてリリースしています。
VBMan ActiveX Controls for Btrieve ver 3.10	2002年3月にPervasive.SQL 2000i SP3に対応して発売されました。また、このバージョンからWindows XPでの動作保証をいたしました。新たにNUMERICSA,NUMERICSTS,CURRENCY, Longvarcharデータ型に対応し、true nullableカラムにも対応しました。

VBMan Controls for Btrieve ver 3.20	2002年10月に発売しました。.NET環境に最小のワークロードで既存のVBアプリを移行したいというお客様のご要望が多かったため、Windows Formsで動作するCOMコンポーネント形式のコントロールも必要と思い、リリースいたしました。名前は依然としてVBManですがこのバージョンからVisual C#でも動作するようになりました。VB6のフォームとWindows Formでは動作の違いが多数ありましたので、内部的にはかなりの変更が必要でした。
VBMan Controls for Btrieve V3.30	2005年11月に発売しました。Pervasive.SQL V8.6 SP3に対応しました。Pervasive.SQLのセキュアデータベースに対応するため、新たにDbLogIn/DbLogOutメソッドを追加しました。DDFのパスとしてURIを指定できるようになりました。
VBMan Controls for Btrieve V3.40	2007年6月に発売しました。Windows Vistaに対応しインストールではocxファイルをシステムにコピーしない仕様に変更しました。また、Pervasive.SQL V9に対応しました。
VBMan Controls for Btrieve v3.50	2022年にVisual Studio 2022でビルドしたメンテナンスリリースです。Visual Studio 2022での動作のため64bit版OCXを追加しました。ネットワーク認証に変更しました。

## Visual Basic.NETへの移行

---

本製品で作成されたVisual Basic 6用アプリケーションはVisual Basic.NETのアップグレードウィザードを使ってVisual Basic.NETアプリケーションに移行可能なことを確認済みです。アップグレードウィザードを使って移行したWindows Formではレイアウトが多少変わってしまうことも確認していますので、アップグレード・ウィザードをご利用後に、フォームのレイアウトや外観をVisual Studio.NETで確認してください。

## VBMan ActiveX Control for Btrieve バージョン 3.Xからの移行

---

コントロールのGUID等に変更してません。OCXファイル名のみ変更になっていますので新しいバージョンをレジストリ登録していただければ既存のプロジェクトの変換は必要ありません。

## VBMan for Btrieve/OCX32 バージョン2.0からの移行

---

ここではVBMan for Btrieve/OCX32を使って作られたVisual Basic アプリケーションをVBMan Control for Btrieve version 3.50を使ったVisual Basicアプリケーションに移行する方法を説明します。この移行説明ではVisual Basic言語が前提であり、VBManの旧バージョンから最

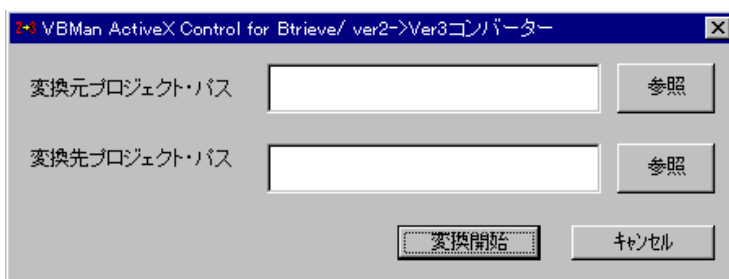
新バージョンへコントロールの入れ替え方法を示すものです。Visual BasicからVisual Basic.NET への変換はVisual Studio.NETで提供されるアップグレードウィザードを使ってください。VBMan の旧バージョンを使ったVisual Basicアプリを最新版のVBMan + Visual Basic.NETアプリへ直接移行することはできません。

変換作業は基本的にはGUIDの変更、ライブラリ名の変更、コントロール・クラス名の変更になります。当製品にはVisual Basicのプロジェクト・ファイル名を指定すると各フォームを変換するツールvbmconv.exeが添付されていますので最初に変換ツールの説明をいたします。

## 変換ツール

変換ツールの使用方法は以下のようになります。

- ① VBMan Controls for Btrieveメニューからver2to3コンバーターを選択して変換ツールを実行します。
- ② 表示されるウィンドウの「変換元プロジェクト・パス」に変換したいVisual Basicプロジェクトファイル（拡張子.vbp）のフル・パスを指定します。
- ③ 「変換先プロジェクト・パス」に変換後のファイルを保存するディレクトリを指定します。「変換後のプロジェクト・パス」には変換元と同じ値を設定することはできませんのでご注意ください。
- ④ 「変換開始」ボタンを押します。
- ⑤ 変換が終了するとウィンドウが終了します。
- ⑥ 変換されたプロジェクトをVisual Basicで読み込み動作を確認します。



## 手動によるフォーム変換

VBMan for Btrieve/OCX32 version 2.00で作成されたVisual BasicのフォームをVBMan Control for Btrieve version 3.5で利用可能なように変更する方法を説明します。フォームの変更にはNotepad.exe等のテキスト・エディターを使って、コントロールのライブラリ名やGUIDを変換することになります。以下は変換手順の概要です。

- ① エディターでVisual Basicのプロジェクト・ファイル(拡張子 .vbp)を開きます。
- ② GUIDを変換します。

変換前の行:

Object={1601CD73-F987-11CE-9262-0080AD156341}#1.0#0; VBMBTR32.OCX

変換後の行:

Object={3235CFE2-D891-11D1-BD2B-004026182472}#1.1#0; VBMBT350.OCX

- ③ Visual Basicのプロジェクト・ファイルを保存します。
- ④ プロジェクトに含まれるフォーム・ファイル(拡張子 .frm)をエディターで開きます。
- ⑤ ライブラリ名を変更します。ライブラリ名はBeginで始まる行のVBMBtrLib.という文字列をVBMBTRLibCtl.に変更します。

変更前の行:

Begin **VBMBTRLib.**VBManBtn VBManBtn6

変更後の行:

Begin **VBMBTRLibCtl.**VBMBtn VBMBtn6

- ⑥ コントロール名を変更します。コントロール名は上記⑤の行にあるライブラリ名の後にドットで表記される文字列です。VBManには13個のコントロールがありますので、後述の変換表を参考に文字列を置換します。
- ⑦ フォーム・ファイルを保存します。

Ver 2.00 コントロール名	Ver 3.x コントロール名
VBMDB	VBManDb
VBManBtn	VBMBtn
VBManEdit	VBMEdit
VBManGrid	VBMGrid
VBManListBox	VBMListBox
VBManComboBox	VBMComboBox
VBManExListBox	VBMExListBox
VBManExComboBox	VBMExComboBox
VBManPic	VBMPic
VBManCheck	VBMCheck
VBManOptBtn	VBMOptBtn
VBManHScroll	VBMHScroll
VBManVScroll	VBMVScroll

コンバージョンツール、手動による変換でもプロパティ設定やご利用のVisual Basicバージョンなどによりコントロールがロードできない場合がありますが、そのような場合はVisual Basicの開発環境で再度コントロールを張っていただくことになります。ご了承ください。

## **VBMan for Btrieve/VBX バージョン1.xからの移行**

---

ここではVBMan for Btrieve/VBX版で作成したアプリケーションをVBMan Controls for Btrieve環境へ移行する際の手順、注意点等を記述します。

### フォームの変換

Visual Basic 2.0Jで作成されたVBManを使ったフォームはそのままではVisual Basic 6.0/5.0/4.0Jでは読み込めません。以下の手順でフォームを移行することになりますが、マイクロソフトによって変更されたVBXとOCXの仕様の変更が原因で完全に移行できる訳ではないことに注意してください。

- ① フォームをテキスト形式で保存  
Visual Basic 2.0Jの「テキスト形式で保存」の機能を使って、移行したいフォームをテキスト形式で保存します。
- ② フォームの編集  
フォームをエディターで開いてVBManのコントロールに「VBMBTRLibCtl.」というモジュールクラスを示すプリフィックスをつけます。以下はVBManデータベース・コントロールにおけるサンプルです。

#### **変更前**

```
Begin VBManDB VBMan1
  Height   = 480
  Left     = 10
  Top      = 480
End
```

#### **変更後**

```
Begin VBMBTRLibCtl.VBManDb VBMan1
  Height   = 480
  Left     = 10
  Top      = 480
End
```

- ③ Visual Basicの起動とコントロールの追加



Visual Basicを起動します。メニューからActiveX Controlを選択して、VBMan Controls for Btrieveをプロジェクトに追加します。

- ④ プロジェクトへフォーム・ファイルの読み込み  
②で編集したフォーム・ファイルをVisual Basicのプロジェクトに追加します。ファイル・メニューの下からファイルの追加を選択し、ファイルの指定ダイアログで②で編集したフォームのファイル名を指定します。
- ⑤ フォームの変換エラーの確認  
フォーム変換ではプロパティの違いなどから、エラーが発生する場合があります。フォームのファイル名と同じ名前で、拡張子が.logのファイルにエラー詳細がレポートされますので、このファイルを参照して、変換できなかったプロパティ等をVisual Basicのデザイン・モードで修正してください。

## データアクセス関数の移行

### バージョン1.0コンパチブル関数

VBManバージョン1.5まではVBXのモジュールにデータベースアクセス関数としてDDFに準拠したBtrieveアクセスを可能とするエントリ・ポイントがエクスポートされていました。これらの関数はVBMan Controls for Btrieveではバージョン1.0コンパチブル関数として同様に関数がエクスポートされています。以前のバージョンではBtrieve.basというファイルにBtrieveのオペレーション・コードと一緒にVisual Basic用に関数の宣言がしてありました。このファイルはVBMan Controls for BtrieveではVBMANV1.BASとして、インストールディレクトリのsamplesディレクトリに存在しますので、このファイルをVisual Basicのプロジェクトに追加してご使用ください。

VBMan Controls for Btrieveではフォームに複数のデータベース・コントロールが設定できるように機能を拡張した結果、このバージョン1.0コンパチブル関数を使う場合は、どのデータベース・コントロールのポジション・ブロックを使うのかを事前に指定する必要があります。コントロールを指定する関数はDbAttachControlで、パラメータはコントロールの名前とそのコントロールが置かれているフォーム名を指定します。以下はサンプル・コードです。

Dim Rc As Integer

```
Rc = DbAttachControl("VBMan1", "Form1")
```

```
If Rc <> 0 Then
```

```
    MsgBox "コンパチブル関数を使うことができません。"
```

```
    Stop
```

```
End If
```

```
Rc = DbAccess(BTR_GET_FIRST, CUST_TABLE, CUST_KEY_PATH)
```

```
If Rc <> 0 Then
```

```
    MsgBox "Btrieve Error " & CStr(rc)
```

```
    Stop
```

```
End If
```

DbAttachControl関数からの戻り値が0でない場合はフォーム名またはコントロール名が正しくないと考えられます。今一度ご確認ください。

## データ・ベース・コントロール・メソッド

Visual Basic 2.0JにおけるVBXの仕様ではカスタム・コントロールに独自のメソッドを追加することはできませんでした。この仕様はActiveX Control(OCX)で拡張され、カスタム・コントロールのデザイナーは独自のメソッドをユーザーに提供することが出来るようになりました。

VBMan Controls for Btrieveでも、このデザイン上の利点を生かすべく、いままでデータ・ベース・アクセス関数として提供されていた関数をデータ・ベース・コントロールのメソッドとして提供するようになりました。単にメソッドとするだけでは、先に紹介したバージョン1.0コンパチブル関数と差別化できないと考え、従来フィールド、テーブル、キーなどは整数のIDで指定する仕様だったのを改め、すべてを文字列でDDF定義時と同じ値を指定可能としました。従来の整数のIDでの方法には、Visual Basic 2.0Jの文字列領域の不足、16bit環境でのパフォーマンス不足、などを考慮したうえでの仕様であり、またループの記述が簡単などの利点もありました。32Bit環境でこれらの制約はある程度緩和されていると考え、パラメータのデザインを変更した訳ですが、お客さまの環境によってはパフォーマンスが不足することなども考えられますので、その場合にはバージョン1.0コンパチブル関数をお使いになられるようお願いいたします。

以下はVBMan Controls for Btrieveのデータベース・アクセス・メソッドの呼び出しコード例です。DbAccess,DbExAccessメソッドの第一パラメータ定義はBtrieve.Basファイルにありますのでご利用ください。

```
Dim rc As Integer
rc = VBMan1.DbAccess( BTR_GET_FIRST, "顧客", "コード")
If rc <> 0 Then
    MsgBox "Btrieveステータス" & CStr(rc)
    Stop
End If
```

### VBMan DDFビルダーの機能

VBMan Controls for Btrieveではインデックスの指定にインデックス名を使えるようになりました。これに対応して同じ先頭フィールド名でセグメント・キーを作成する場合、インデックスを識別するために、インデックス名が設定可能になっています。「インデックス」メニューに「インデックス名の追加」、「インデックス名の削除」が追加されています。

フォームの生成に関しては以前のVBMan for Btrieve/VBXをVB2で利用できるフォームとVBMan Controls for Btrieveバージョン3.5をVisual Basic 6.0Jで利用できるフォーム生成が

可能です。Visual Basic .NETで使うフォームは生成できませんが、Visual Basic 6用のフォームはVisual Studio.NETのアップグレードウィザードでVisual Basic.NET用に変換できます。

# カスタム・コントロール・リファレンス

---

ここでは、VBMan Controls for Btrieveで利用可能な13個のカスタム・コントロールについて説明します。最初にフォームに設定するコントロールは次に説明するデータベース・コントロールです。この章の説明では各コントロールの概要、使用上の注意などを記述します。カスタム・プロパティ、カスタム・イベント、カスタム・メソッドの詳細はそれぞれのリファレンスを参照してください。

## データベース・コントロール

---

### 概要

VBManデータベース・コントロールはVBManを使用するアプリケーションにおいて、フォーム等を作成する場合に最初に設定するコントロールです。このコントロールのDDFDirプロパティで設定したディレクトリ情報からDDFを読み込みます。

このコントロールがフォームに設定されていない場合は、他のVBManコントロールがフォームに存在しても、DbSource,DbTableプロパティ等が指定可能となりませんのでご注意ください。

VBManデータベース・コントロールはデザイン時にはアイコン表示されますが、実行時には不可視になります。デザイン時にはフォームの設計で邪魔にならない適当な場所に設定します。

VBManデータベース・コントロールが読み込むDDF定義に存在するテーブルについて、テーブルに関連するBtrieveファイルをオープンした時には、このコントロール単位でBtrieveのポジション・ブロックとレコード・イメージを保持するバッファを管理します。たとえば、同じファイルを別々のポジション・ブロックで管理したい場合は同じフォームに複数のVBManデータベース・コントロールを設定することで可能になります。

VBManデータベース・コントロールをVisual Basicで使う場合、コントロール配列にして使うことはできません。他のVBManコントロールはDbSource,DbListSourceプロパティにVBManデータベース・コントロールのNameプロパティの値を参照する仕様の為、コントロール配列にするとNameプロパティで識別できなくなります。

以下はVBManデータ・ベース・コントロールに特有なプロパティ、イベント、メソッドの一覧です。

### カスタム・プロパティ

---

CacheDDF	DDFDir	DistName	OpenMode	OwnerName
----------	--------	----------	----------	-----------

## カスタム・イベント

---

BtrieveError

## カスタム・メソッド

---

DbAbortTransaction	DbAccess
DbAccessByID	DbBeginConcurTransaction
DbBeginTransaction	DbClearControlData
DbClearFieldBuffer	DbClose
DbCloseAll	DbCreate
DbEndTransaction	DbExAccess
DbExFree	DbExGetFieldData
DbExGetPosition	DbExSetMode
DbFindPercentage	DbGetByPercentage
DbGetDataSize	DbGetDataType
DbGetDirect	DbGetFieldData
DbGetFieldDataByID	DbGetFieldName
DbGetIndexName	DbGetModulePath
DbGetNumOfField	DbGetNumOfIndex
DbGetNumOfRecords	DbGetNumOfTable
DbGetPosBlock	DbGetPosition
DbGetRecord	DbGetRecordImage
DbGetRecordLength	DbIsOpen
DbLoadDDF	DbOpen
DbOpenAll	DbReset
DbSetFieldData	DbSetFieldDataByID
DbSetFileName	DbSetLockBias
DbSetRecord	DbSetRecordImage
DbTransferFromControl	DbTransferToControl
DbUnlock	

## ボタン・コントロール

---

概要

VBManボタン・コントロールはユーザーのマウス・クリックによるボタンの押下時に DbOperationプロパティに設定したBtrieveオペレーションを実行します。

3

登録系のオペレーションを設定した場合には、同じフォームにある、このコントロールを指定するようにDbSource,DbTableプロパティに値が設定してあるVBManコントロールからデータを収集して、DDFにしたがってレコード・イメージを作成し、Btrieveのオペレーションを発行します。

4

参照系のオペレーションがDbOperationプロパティに設定されている場合は、同じフォームにある、このコントロールを指定するようにDbSource,DbTableプロパティが指定されたコントロールにレコード・イメージのバッファからBtrieveから読み込んだデータを設定します。

DbOperationプロパティとして設定できるのは以下の項目です。これらはBtrieveの機能をそのまま実現しています。

First	最少の値をもつキーに該当するレコードを得る。
Last	最大の値をもつキーに該当するレコードを得る。
Insert	新規のレコードを追加。
Delete	現在のレコードを削除。
Update	現在のレコードを更新。
Next	次の値をもつキーに該当するレコードを得る。
Previous	前の値をもつキーに該当するレコードを得る。
GetEqual	一致するキーに該当するレコードを得る。
GetGreater	指定したキーより大きな値をもつキーに該当するレコードを得る。
GetGreaterOrEqual	指定したキーと同じまたは、より大きな値で一番近いをもつキーに該当するレコードを得る。
GetLessThan	指定したキーより小さな値をもつキーに該当するレコードを得る。

3

Insert,Updateオペレーション。

4

GetEqual,GetNextオペレーションなど。

GetLessThanOrEqual	指定したキーより同じ、または、より小さな値で一番近い値をもつキーに該当するレコードを得る。
StepFirst	物理先頭レコードを得る。
StepLast	物理最終レコードを得る。
StepPrevious	前の値をもつレコードを得る。
StepNext	次の値をもつレコードを得る。
InsertNCC	新規レコードを追加。論理カレンシー情報は設定しない。
UpdateNCC	レコードを更新。論理カレンシー情報は設定しない。

5

以下の表はVBManボタンに特有のプロパティ、イベント、メソッドをあらわします。

#### カスタム・プロパティ

---

Abort	ClearControlData	ConfirmMsg	DbIndex
DbLocks	DbOperation	DbTable	DbSource

#### カスタム・イベント

---

BtrieveError    SetData

#### カスタム・メソッド

---

なし

## エディット・コントロール

---

### 概要

VBManテキストコントロールはDbSource,DbTable,DbFieldプロパティによって指定されるBtrieveデータベースのフィールドのデータを表示、登録します。Btrieveとのデータの交換はVBManボタン・コントロールをクリックした場合や、VBManスクロールを使った場合、DbTransferToControl, DbTransferFromControlメソッド/関数を使った場合におこなわれます。

---

<sup>5</sup> 対応するVisual BasicまたはWindowsのコントロールと比べた場合に特有のプロパティを指します。

## 入カマスクについて

VBManエディット・コントロールは関係するフィールドのデータタイプがDate型またはTime型の場合で、Formatプロパティが指定されていない場合はキー入力をマスクして入力します。Date型の場合はVBMAN.INIファイルの設定によって、yyyy/mm/dd,yy/mm/ddの形式を選択可能です。Time型についてはhh:mm:ss形式固定です。これらのマスクはコントロールからフォーカスがなくなる時におこなわれます。これらのデフォルト入カマスク以外のマスクをしたい場合はFormatプロパティとFormatOptionプロパティを組み合わせ設定することになります。Format,FormatOptionプロパティの詳細についてはカスタム・プロパティ・リファレンスを参照してください。

以下の表はVBManエディットに特有のプロパティ、イベント、メソッドです。

### カスタム・プロパティ

---

DbField	DbSource	DbTable	Format
FormatOption	NumericMask	ReadOnly	UpperCase

### カスタム・イベント

---

なし

### カスタム・メソッド

---

なし

## スクロール・コントロール

---

VBMan水平・垂直スクロール・コントロールにより、レコードの移動ができます。レコードの移動はDbIndexで指定されるキー順となります。キーなしのBtrieveファイル、Btrieve 5.x形式のBtrieveファイルではこのコントロールは使えません。

また、このコントロールはVBManボタンにカレント・レコードを変えるオペレーションを設定した場合はこれらのコントロールに連動してキャレット・ポジションを変更します。

レコード件数が変わるようなオペレーションを発行した場合、Refreshメソッドを呼び出すことで、スクロール・コントロールのキャレット位置を正しい位置に設定することができます。

---

<sup>6</sup> Btrieveのパーセンテージ・オペレーションを使っているため。



以下の表はVBManスクロール・コントロールに特有のプロパティ、イベント、メソッドです。

---

#### カスタム・プロパティ

---

AutoFirst	DbIndex	DbSource	DbTable
-----------	---------	----------	---------

---

#### カスタム・イベント

---

BtrieveError

---

#### カスタム・メソッド

---

なし

---

## チェック・ボックス・コントロール

---

### 概要

VBManチェック・ボックス・コントロールはDbSource, DbTable, DbFieldプロパティで指定されるデータベースのフィールドの値がDbValueTrueプロパティで指定される値と同一になった場合にチェックされます。

このデータ・ベースのフィールドの値とは、VBManボタン・コントロールや、スクロール・コントロールでカレント・レコードとしてポジショニングされたレコードにおけるフィールドの値です。データ・ベース・アクセス・メソッドまたはバージョン1.xコンパチブル・関数でカレント・レコードをポジショニングした場合は、DbTransferToControlメソッドなどでデータをカスタム・コントロールに転送した時にフィールドの値が検査され、値が同一であれば、チェックされます。

以下の表はVBManチェック・ボックス・コントロールに特有のプロパティ、イベント、メソッドです。

---

#### カスタム・プロパティ

---

DbField	DbSource	DbTable	DbValueFalse
DbValueIndeterminate	DbValueTrue		

---

#### カスタム・イベント

---

なし

---

#### カスタム・メソッド

---

なし

## オプション・ボタン・コントロール

---

### 概要

VBManオプション・ボタンコントロールはDbSource, DbTable, DbFieldプロパティで指定されるデータベースのフィールドの値がDbValueTrueプロパティで指定される値と同一になった場合にチェックされます。

このデータ・ベースのフィールドの値とは、VBManボタン・コントロールや、スクロール・コントロールでカレント・レコードとしてポジショニングされたレコードにおけるフィールドの値です。データ・ベース・アクセス・メソッドまたはバージョン1.xコンパチブル・関数でカレント・レコードを位置づけした場合は、DbTransferToControlメソッドなどでデータをカスタム・コントロールに転送した時にフィールドの値が検査され、値が同一であれば、チェックされます。

以下の表はVBManオプション・ボタン・コントロールに特有のプロパティ、イベント、メソッドです。

### カスタム・プロパティ

---

DbField	DbSource	DbTable	DbValueFalse
DbValueTrue			

### カスタム・イベント

---

なし

### カスタム・メソッド

---

なし

## リスト・ボックス・コントロール

---

### 概要

VBManデータベース・リストボックス・コントロールはプロパティ

**DbListField,DbListSource,DbListTable**の値で指定されたカラムのデータすべてをデータベースから取得し表示します。データ更新のタイミングは**DbUpdateOption**プロパティの設定により、なし、初期化時、フォーカス時を選択できます。

また、DbField,DbSource,DbTableプロパティで指定されるフィールドのデータとして、このリスト・ボックスの選択値を設定、参照ができます。データを設定、参照するのはVBManボタンの登録系のオペレーションまたは、DbTransferToControl,DbTransferToControlメソッドです。

VBManリスト・ボックスはDbListIndexプロパティで指定されるキー・パスで、BtrieveのGET FIRSTオペレーションを発行し、ファイルの最後までGET NEXTオペレーションを発行して、リスト・ボックスにデータを挿入します。SortプロパティをTrueに設定している場合はVBManリスト・ボックスがメモリでソートしてデータを表示し、DbListIndexプロパティで指定されるキー順にはならないこともあるので注意してください。

VBManリスト・ボックスは、レコードをリストに表示する直前にFormatイベントが発生します。このイベントのパラメータはこれから表示しようとするフィールドの値です。このイベント内ではこの表示するデータを含むレコードがカレント・レコードになっています。したがって、DbGetFieldDataメソッドでフィールドの値を取得可能です。

Formatイベントでは他のフィールドの値を検査、計算した結果、別の文字列を表示したければ、Formatイベントのパラメータで渡る文字列に表示を更新する文字列を設定することで可能です。Formatイベント内でイベントのパラメータにヌル文字列を設定した場合、そのレコードはリスト・ボックスに表示されません。以下はVisual Basicでのサンプル・コードです。

‘マイナスのデータは表示しない例

```
Sub VBManListBox1_Format( LineData As String )
If Left$(LineData,1) = "-" Then
  LineData = ""      ‘ヌル文字を設定
End If
End Sub
```

また、現在のレコードまででリストを止めたい場合はAbortプロパティにTrue値を設定します。

Visual Basicをコントロール・コンテナにした場合は、DbUpdateOptionプロパティを初期化時に設定すると、Formatイベントが発生しません。これは初期化時にはVisual Basicがイベントの発生を禁止していることが原因です。Formatイベントを正常に発行させるにはフォームの ActivateイベントでこのコントロールのRefreshメソッドをよびだすことで回避できます。<sup>7</sup>

VBManリスト・ボックスの内容を再度Btrieveデータベースから読み込みたい場合は、Refreshメソッドを使います。

VBManリスト・ボックスに保持できる最大行数は32767件です。この値は32bitのリスト・ボックスの仕様に依存する行数です。

---

<sup>7</sup> サンプル・コードは巻末のプログラミングのヒントにあります。

第2パラメータは文字列を挿入する位置ですが、-1を指定するとリストの最後に追加します。

以下の表はVBManリスト・ボックスに特有のプロパティ、メソッド、イベントです。

#### カスタム・プロパティ

---

Abort	DbField	DbListField	DbListIndex
DbListSource	DbListTable	DbSource	DbTable
DbUpdateOption			

#### カスタム・イベント

---

Format

#### カスタム・メソッド

---

なし

### コンボ・ボックス・コントロール

---

#### 概要

VBManデータベース・コンボ・ボックス・コントロールはプロパティ**DbListField,DbListSource,DbListTable**の値で指定されたカラムのデータすべてをデータベースから取得し表示します。データ更新のタイミングは**DbUpdateOption**プロパティの設定により、なし、初期化時、フォーカス時を選択できます。

また、DbField,DbSource,DbTableプロパティで指定されるフィールドのデータとして、このコンボ・ボックスの選択値を設定、参照ができます。データを設定、参照するのはVBManボタンの登録系のオペレーションまたは、DbTransferToControl,DbTransferToControlメソッドです。VBManコンボ・ボックスはDbListIndexプロパティで指定されるキー・パスで、BtrieveのGET FIRSTオペレーションを発行し、ファイルの最後までGET NEXTオペレーションを発行して、コンボ・ボックスにデータを挿入します。SortプロパティをTrueに設定している場合はVBManコンボ・ボックスがメモリでソートしてデータを表示し、DbListIndexプロパティで指定されるキー順にはならないこともあるので注意してください。

VBManコンボ・ボックスは、レコードをリストに表示する直前にFormatイベントが発生します。このイベントのパラメータはこれから表示しようとするフィールドの値です。このイベント内ではこの表示しようとするデータを含むレコードがカレント・レコードになっています。したがって、DbGetFieldDataメソッドでフィールドの値を取得可能です。

Formatイベントでは他のフィールドの値を検査、計算した結果、別の文字列を表示したければ、Formatイベントのパラメータで渡る文字列に表示を更新する文字列を設定することで可能です。Formatイベント内でイベントのパラメータにヌル文字列を設定した場合、そのレコードはコンボ・ボックスに表示されません。

また、現在のレコードまででリストを止めたい場合はAbortプロパティにTrue値を設定します。

Visual Basicをコントロール・コンテナにした場合は、DbUpdateOptionプロパティを初期化時に設定すると、Formatイベントが発生しません。これは初期化時にはVisual Basicがイベントの発生を禁止していることが原因です。Formatイベントを正常に発行させるにはフォームのActivateイベントでこのコントロールのRefreshメソッドをよびだすことで回避することができます。

VbManコンボ・ボックスの内容を再度Btrieveデータベースから読み込みたい場合は、Reloadメソッドを使います。

VbManコンボ・ボックスに保持できる最大行数は32767件です。この値は32bitのコンボ・ボックスの仕様に依存する行数です。

以下の表はVbManコンボ・ボックスに特有のプロパティ、メソッド、イベントです。

#### カスタム・プロパティ

---

Abort	DbField	DbListField	DbListIndex
DbListSource	DbListTable	DbSource	DbTable
DbUpdateOption			

#### カスタム・イベント

---

Format

#### カスタム・メソッド

---

Reload

### 概要

VBManピクチャー・コントロールはグラフィック・イメージ・データを表示するコントロールです。Visula Basicに標準のイメージ・コントロールと同等の機能にデータベースからのデータを表示する機能を追加しました。

表示可能なイメージ・データは以下の形式のファイルです。

- | ビットマップ (.bmp)
- | アイコン (.ico)
- | ウィンドウズ・メタファイル (.wmf)

VBManピクチャー・コントロールではDbSource,DbTable,DbFieldプロパティで指定される文字列型データベース・フィールドの値をファイル名としてディスクからイメージ・データを読み込み表示します。

指定されたファイルのイメージ・データタイプの判別にはファイル名の拡張子を使用します。拡張子が指定されない場合はビットマップ(.bmp)と仮定します。

BlobPathプロパティにはイメージ・データが存在するディスク上のパスを指定することが可能です。この場合、データベース・フィールドの値はファイル名だけが指定されていることが必要です。BlobPathプロパティに指定がなく、データベース・フィールドにファイル名だけが指定されるような場合はカレント・ドライブのカレント・ディレクトリにファイルが存在する扱いになります。

VBManピクチャー・コントロールに再度データを読み込む場合はRefreshメソッドを呼び出します。

以下の表はVBManピクチャー・コントロールに特有のプロパティ、メソッド、イベントです。

### カスタム・プロパティ

---

BlobPath	DbField	DbSource	DbTable
----------	---------	----------	---------

### カスタム・イベント

---

なし

### カスタム・メソッド

---

なし

### 概要

VBMan拡張リスト・ボックス・コントロールはBtrieveのExtendedオペレーションに対応したリスト・ボックス・コントロールです。BtrieveのExtendedオペレーションの特徴は以下です。

- | 複合検索が可能
- | 複数レコードを一度に転送するためパフォーマンスが高い
- | フィールドの選択によりネットワークの負荷を軽減

通常のBtrieve呼び出しではパラメータの構成が非常に複雑ですが、VBManではプロパティを指定するだけで簡単にExtendedオペレーションが使用可能です。

### プロパティの設定

Extended系オペレーションを実行するためのプロパティ設定の手順を説明します。

- ① DbListSourceプロパティの指定
- ② DbListTableプロパティの指定  
検索対象とするテーブル名を指定します。
- ③ DbExOperationプロパティの指定  
GET PREVIOUS EXTENDED, GET NEXT EXTENDED, STEP PREVIOUS EXTENDED, STEP NEXT EXTENDEDの中から選択します。
- ④ DbSkipRecordsプロパティの指定  
検索条件を満たさない場合にスキップするレコード数を設定します。0を指定するとシステム定義の最大値をBtrieveが選択して使用します。Btrieve 6.15では4,095です。
- ⑤ DbMaxRecordsプロパティの指定  
検索結果の最大レコード件数を設定します。Btrieveでは結果はリニアなメモリに返されるため、65,536バイト以上を越えることができません。65,536をDbFieldsプロパティで指定する抽出するフィールドの合計サイズで割った値以上は設定できません。
- ⑥ DbListFieldsプロパティの指定  
抽出するフィールド名をカンマで区切った文字列を指定します。フィールドIDでも指定可能です。フィールドIDはDDFビルド-の表示値を0オフセットにした値です。(例：1,2,4)
- ⑦ DbConditionsプロパティの指定  
検索条件を指定します。指定の詳細はカスタム・プロパティ・リファレンスを参照してください。
- ⑧ DbTabStopsプロパティの指定  
VBManリスト・ボックスではFormatイベントに渡されるデータは1つのフィールドの値でしたが、VBMan拡張リスト・ボックスの場合は複数の項目が渡されます。DbTabStopsプロパティを設定した場合はFormatイベントにはDbFieldsで指定された複数のデータの区切りはタブ(&H09)文字になります。DbTabStopsプロパティに何も指定

しない場合はスペース文字(&H20)が区切り文字になります。DbTabStopsプロパティは半角の文字数をカンマで区切って指定します。(例："10,20,30,40")DbTabStopsプロパティは実行時のコントロールを作成する時に一度だけ参照されます。実行時にダイナミックに変更することはできません。

⑨ DbLeaveResultsプロパティの指定

このプロパティの設定は必須ではありません。検索の結果をメモリに保持しておき、DbExGetFieldData関数などを使って取り出す場合にこのプロパティをTrueに設定します。DbExGetFieldData関数に渡す結果IDはDbResultIDプロパティに保持されます。(ランタイムのみのプロパティのため、プロパティ・リスト・ボックスには表示されないことに注意してください) DbUpdateOptionをフォーカス時に設定した場合はDbExFreeでメモリを解放するコードを記述することが困難と思われるのでお勧めしません。

### 検索開始レコード

VBMan拡張リスト・ボックスではすべてのレコードを検索対象とするため、Extendedオペレーションの前にカレント・レコードのポジショニングを実行します。各オペレーションとポジショニングの関係を以下に示します。

EXTENDED オペレーション	POSITIONINGオペレーション
GET PREVIOUS EXTENDED	GET LAST
GET NEXT EXTENDED	GET FIRST
STEP PREVIOUS EXTENDED	STEP LAST
STEP NEXT EXTENDED	STEP FIRST

### 注意事項

Refreshメソッドを使うとBtrieveデータベースから最新のデータを取得し、リスト・ボックスの内容を更新することができます。

以下の表はVBManに特有のプロパティ、イベント、メソッドです。

### カスタム・プロパティ

---

Abort	DbCompareCase	DbConditions	DbExOperation
DbFields	DbLeaveResults	DbListFields	DbListIndex
DbListSource	DbListTable	DbLocks	DbMaxRecords
DbIndex	DbResultID	DbSkipRecords	DbTable
DbTabStops	DbUpdateOption	SavePosition	



## カスタム・イベント

---

BtrieveError      Format

## カスタム・メソッド

---

Reload

## 拡張コンボ・ボックス・コントロール

---

### 概要

VBMan拡張コンボ・ボックス・コントロールはBtrieveのExtendedオペレーションに対応したコンボ・ボックス・コントロールです。BtrieveのExtendedオペレーションの特徴は以下です。

- ┆ 複合検索が可能
- ┆ 複数レコードを一度に転送するためパフォーマンスが高い
- ┆ フィールドの選択によりネットワークの負荷を軽減

通常のBtrieve呼び出しではパラメータの構成が非常に複雑ですが、VBManではプロパティを指定するだけで簡単にExtendedオペレーションが使用可能です。

### プロパティの設定

Extendedオペレーションを実行させるためのプロパティ設定の手順を説明します。

- ① DbListSourceプロパティの指定
- ② DbListTableプロパティの指定  
検索対象とするテーブル名を指定します。
- ③ DbExOperationプロパティの指定  
GET PREVIOUS EXTENDED, GET NEXT EXTENDED, STEP PREVIOUS EXTENDED, STEP NEXT EXTENDEDの中から選択します。
- ④ DbSkipRecordsプロパティの指定  
検索条件を満たさない場合にスキップするレコード数を設定します。0を指定するとシステム定義の最大値をBtrieveが選択して使用します。Btrieve 6.15では4,095です。
- ⑤ DbMaxRecordsプロパティの指定  
検索結果の最大レコード件数を設定します。Btrieveでは結果はリニアなメモリに返されるため、65,536バイト以上を越えることができません。65,536をDbFieldsプロパティで指定する抽出するフィールドの合計サイズで割った値以上は設定できません。

- ⑥ DbListFieldsプロパティの指定  
抽出するフィールド名をカンマで区切った文字列を指定します。フィールドIDでも指定可能です。フィールドIDはDDFビルダ<sup>®</sup>の表示値を0オフセットにした値です。(例：1,2,4)
- ⑦ DbConditionsプロパティの指定  
検索条件を指定します。指定の詳細はカスタム・プロパティ・リファレンスを参照してください。
- ⑧ DbDelimitCharプロパティの指定  
VBMan拡張コンボ・ボックスでは項目間の区切り文字をDbDelimitCharプロパティで指定します。このプロパティに何も指定しない場合はスペースが項目間の区切り文字になります。
- ⑨ DbLeaveResultsプロパティの指定  
このプロパティの設定は必須ではありません。検索の結果をメモリに保持しておき、DbExGetFieldData関数などを使って取り出す場合にこのプロパティをTrueに設定します。DbExGetFieldData関数に渡す結果IDはDbResultIDプロパティに保持されます。(ランタイムのみのプロパティのため、プロパティ・リスト・ボックスには表示されないことに注意してください) DbUpdateOptionをフォーカス時に設定した場合はDbExFreeでメモリを解放するコードを記述することが困難と思われるのでお勧めしません。

## 検索開始レコード

VBMan拡張コンボ・ボックスではすべてのレコードを検索対象とするため、Extendedオペレーションの前にカレント・レコードのポジショニングを実行します。各オペレーションとポジショニングの関係を以下に示します。

EXTENDED オペレーション	POSITIONINGオペレーション
GET PREVIOUS EXTENDED	GET LAST
GET NEXT EXTENDED	GET FIRST
STEP PREVIOUS EXTENDED	STEP LAST
STEP NEXT EXTENDED	STEP FIRST

## 注意事項

Refreshメソッドを使うとBtrieveデータベースから最新のデータを取得し、コンボ・ボックスの内容を更新することができます。

以下の表はVBMan拡張コンボ・ボックスに特有のプロパティ、メソッド、イベントです。

### カスタム・プロパティ

---

Abort	DbCompareCase	DbConditions	DbDelimitChar
DbExOperation	DbFields	DbLeaveResults	DbListFields

DbListIndex	DbListSource	DbListTable	DbLocks
DbMaxRecords	DbIndex	DbResultID	DbSkipRecords
DbTable	DbUpdateOption	SavePosition	

## カスタム・イベント

---

BtrieveError      Format

## カスタム・メソッド

---

Reload

## グリッド・コントロール

---

### 概要

VBManグリッド・コントロールはBtrieveのExtendedオペレーションを使ってグリッドにデータを表示します。

### プロパティの設定

Extendedオペレーションを実行させるためのプロパティ設定の手順を説明します。

- ① DbListSourceプロパティの指定
- ② DbListTableプロパティの指定  
検索対象とするテーブル名を指定します。
- ③ DbExOperationプロパティの指定  
GET PREVIOUS EXTENDED, GET NEXT EXTENDED, STEP PREVIOUS EXTENDED, STEP NEXT EXTENDEDの中から選択します。
- ④ DbSkipRecordsプロパティの指定  
検索条件を満たさない場合にスキップするレコード数を設定します。0を指定するとシステム定義の最大値をBtrieveが選択して使用します。Btrieve 6.15では4,095です。
- ⑤ DbMaxRecordsプロパティの指定  
検索結果の最大レコード件数を設定します。Btrieveでは結果はリニアなメモリに返されるため、65,536バイト以上を越えることができません。65,536をDbFieldsプロパティ

で指定する抽出するフィールドの合計サイズで割った値以上は設定できません。

- ⑥ DbListFieldsプロパティの指定  
抽出するフィールド名をカンマで区切った文字列を指定します。フィールドIDでも指定可能です。フィールドIDはDDFビルダ<sup>®</sup>の表示値を0オフセットにした値です。(例：1,2,4)
- ⑦ DbConditionsプロパティの指定  
検索条件を指定します。指定の詳細はカスタム・プロパティ・リファレンスを参照してください。
- ⑧ DbLeaveResultsプロパティの指定  
このプロパティの設定は必須ではありません。検索の結果をメモリに保持しておき、DbExGetFieldData関数などを使って取り出す場合にこのプロパティをTrueに設定します。DbExGetFieldData関数に渡す結果IDはDbResultIDプロパティに保持されます。(ランタイムのみのプロパティのため、プロパティ・リスト・ボックスには表示されないことに注意してください) DbUpdateOptionをフォーカス時に設定した場合はDbExFreeでメモリを解放するコードを記述することが困難と思われるのでお勧めしません。

## 検索開始レコード

VBManグリッドではすべてのレコードを検索対象とするため、Extendedオペレーションの前にカレント・レコードのポジショニングを実行します。各オペレーションとポジショニングの関係を以下に示します。

EXTENDED オペレーション	POSITIONINGオペレーション
GET PREVIOUS EXTENDED	GET LAST
GET NEXT EXTENDED	GET FIRST
STEP PREVIOUS EXTENDED	STEP LAST
STEP NEXT EXTENDED	STEP FIRST

## 注意事項

### Btrieveファイル・バージョン

Btrieveのファイル・フォーマットはバージョン6.x以上であることが必要です。

### 表示内容の更新

Refreshメソッドを使うとBtrieveデータベースから最新のデータを取得し、グリッドを更新することができます。

### トランザクション・コントロール

AllowDelete,AllowUpdateプロパティをTrueに設定した場合、表示データを削除、修正することが可能です。グリッドのDbListSourceで指定したデータ・ベース・コントロールのDbBeginTransaction/DbAbortTransaction/DbEndTransactionメソッドを適宜発行すること

で、表示データの修正をトランザクションとしてコントロールすることが出来ます。

以下の表はVBMangリッドに特有のプロパティ、メソッド、イベントです。

#### カスタム・プロパティ

---

Abort	AlignFixedCells	AllowDelete	AllowUpdate
AllowUserResizing	CellMaxLength	DbCompareCase	DbConditions
DbExOperation	DbLeaveResults	DbListFields	DbListIndex
DbListSource	DbListTable	DbLocks	DbMaxRecords
DbResultID	DbSkipRecords	DbUpdateOption	SetColNames
SetColWidth	SetRecordNumbers		

#### カスタム・イベント

---

AfterColUpdate	AfterDelete	AfterUpdate	BeforeColUpdate
BeforeDelete	BeforeUpdate	BtrieveError	Format

#### カスタム・メソッド

---

Reload

## カスタム・プロパティ・リファレンス

---

VBMan Controls for Btrieveのカスタム・プロパティについての解説です。

### Abort

---

#### 概要

AbortプロパティはVBManボタンのSetDataイベントの中でデータベースに登録するデータを検査して不都合を発見した場合、値をTrueにセットすることで、データベースへの操作を中止できます。

また、リスト・ボックス等のFormatイベント中でTrueに設定された場合には、リストへのデータの読み込みは中断されます。

#### プロパティ・データ・タイプ

文字列型

#### 対象コントロール

ボタン、リスト、コンボ、拡張リスト、拡張コンボ、グリッド

### AlignFixedCells

---

#### 概要

グリッドの固定セル上のテキストの右アライメントの有効/無効を設定します。設定値は以下となります。

プロパティ値	意味
False	固定セルのテキストの右アライメントを無効とする。固定セルのテキストは常に左アライメントとなる。
True	固定セルのアライメントはTextAlignプロパティの設定が有効となる。

#### プロパティ・データ・タイプ

論理型

#### 対象コントロール

グリッド

### 概要

VBManデータベース・コントロールが実行モードでロードされたときにこのプロパティに値1が設定されていると、DDFDirプロパティで指定されるDDF定義にあるテーブルに関連するBtrieveファイルをすべてオープンします。このプロパティに値0が指定されている場合は、各VBManコントロールがBtrieveファイルにアクセスすることが必要となるイベントが発生した時にBtrieveファイルをオープンします。

### プロパティ・データ・タイプ

整数型

### 対象コントロール

データベース

### AllowDelete

---

### 概要

グリッドに表示されているデータの削除を許可します。設定値は以下のようになります。

プロパティ値	意味
False	削除は実行されません。
True	削除は実行されます。

### プロパティ・データ・タイプ

論理型

### 対象コントロール

グリッド

### 注意

グリッドからデータを削除する場合はFixedRowsプロパティに1以上を設定してマウスで削除する行を指定できるようにする必要があります。削除の操作手順は以下のようになります。

- ① 削除したい行の左固定セルをマウス左クリックで選択します。
- ② 削除キーを押します。

### AllowUpdate

---

## 概要

グリッドに表示されているデータの更新を許可します。設定値は以下のようになります。

プロパティ値	意味
False	更新は実行されません。
True	更新は実行されます。

## プロパティ・データ・タイプ

論理型

## 対象コントロール

グリッド

AllowUserResizing

---

## 概要

グリッド・セルのリサイズを指定します。設定値は以下のようになります。

プロパティ値	意味
0	リサイズしない
1	カラム方向にリサイズ
2	ロー方向にリサイズ
3	両方向にリサイズ

## プロパティ・データ・タイプ

正数型

## 対象コントロール

グリッド

AutoFirst

---

## 概要

このプロパティにTrueを設定すると、スクロール・バー・コントロールがロードされた時に BtrieveのGET FIRSTオペレーションを実行します。GET FIRSTオペレーションの結果はVBMan



エディット・コントロールなどに表示されます。VBManリスト・ボックス、コンボ・ボックスを併用する場合は、フォームにコントロールを配置する順に依存してリスト・ボックスにデータが表示されるタイミングとスクロール・バーがGET FIRSTオペレーションを実行するタイミングが前後することがありますのでご注意ください。スクロールとデータ登録関連のプロパティをセットしたリスト・ボックス、コンボ・ボックスを併用する場合は、Visual Basicでしたら、FormのActivateイベントでリスト、スクロールの順で明示的にRefreshメソッドを呼び出すコードを記述してください。以下はサンプル・コードです。Staticに用意されているフラグはActivateイベントを初期化時の1回目だけにRefreshを実行するための識別に使っています。

```
Sub Form1_Activate  
Static bFirstFlag As Integer
```

```
If bFirstFlag = False Then  
    VBManListBox1.Refresh  
    VBManVScroll1.Refresh  
    bFirstFlag = True  
End If
```

### **プロパティ・データ・タイプ**

論理型

### **対象コントロール**

水平スクロール・バー、垂直スクロール・バー

## BlobPath

### **概要**

VBManピクチャー・コントロールでイメージ・データの存在するディスク上のパスを指定します。このプロパティを指定しない場合はVBManピクチャー・コントロールはカレント・ドライブのカレント・ディレクトリに存在するイメージ・ファイルを表示します。

### **プロパティ・データ・タイプ**

文字列型

### **対象コントロール**

ピクチャー・コントロール

## CacheDDF

### **概要**

このプロパティの値をTrueに設定した場合、同一プロセス内に同じDDFDirプロパティを定義したデータ・ベース・コントロールが存在し、すでにDDF情報を読みこんでいる場合に限り、実際のDDFを再度読みこむのではなく、メモリ間のコピーによってDDF情報を複製してパフォーマンスの向上を実現します。プロパティの値をFalseと設定した場合にはメモリでのDDF複製は行われず、DDFDirプロパティで指定されるDDFを読みこんでDDF情報を設定します。

### プロパティ・データ・タイプ

論理型

### 対象コントロール

データ・ベース

---

## CellMaxLength

### 概要

AllowUpdateプロパティをTrueに設定してデータの更新を可能としたグリッドでカラム単位で編集ウィンドウに入力できる最大バイト数を設定します。カラムはプロパティのインデックスとして指定します。

### プロパティ・データ・タイプ

論理型

### 対象コントロール

グリッド

### 注意

設定する値は文字の最大バイト数で、文字数では無いことに注意してください。漢字は2バイトとしてカウントされます。

### サンプル・コード

```
VBMGrid1.CellMaxLength(2) = 32 ' 32バイトまで文字を入力可能
```

---

## ClearControlData

### 概要

このプロパティの値をTrueに設定すると、DbOperationプロパティにInsert,Delete,Updateオペレーションを設定した場合に、ボタンのクリックによりBtrieveの呼び出しが正常終了した後、VBManエディットなどデータが表示されていたフィールドのデータを空白にします。データを空白にするコントロールはDbSource,DbTableプロパティが一致しているコントロールに限定

されます。

### プロパティ・データ・タイプ

文字列型

### 対象コントロール

ボタン

## ConfirmMsg

---

### 概要

ConfirmMsgプロパティを設定すると、VBManボタンに設定されたオペレーションをBtrieveデータベースに対して実行する前に確認メッセージが表示されます。確認メッセージにNoと答えた場合はデータベースに対するオペレーションは実行されません。このプロパティは例えば、削除のオペレーションなどをボタンに設定し、削除する前にユーザーに確認を促したい場合に使います。

### プロパティ・データ・タイプ

文字列型

### 対象コントロール

ボタン

## DbCompareCase

---

### 概要

BtrieveのExtended系を使用するVBManコントロールでこのプロパティ値をTrueに設定すると、DbConditionsプロパティで指定される検索条件を比較するとき英数字のケース・センスを指定できます。このプロパティの設定はそのままBtrieveのExtended呼び出しのパラメータ・ブロックに設定されます。

プロパティ値	動作
0	大文字・小文字を区別
1	大文字・小文字を区別しない

### プロパティ・データ・タイプ

整数型

### 対象コントロール

## DbConditions

---

### 概要

BtrieveのExtended系を使用するVBManコントロールにおいて、検索条件を指定します。検索条件の指定は以下のような書式です。

#### フィールドの指定

フィールドは@をプリフィックスとして指定します。たとえば@1はフィールドIDの1を意味します。VBMan for Btrieve/VBXプロフェッショナル版ではフィールドIDしか指定できませんでしたが、VBMan Controls for Btrieveではフィールド名をそのまま指定できます。

#### 比較演算子

比較演算子は以下の値を指定可能です。

=	等しい
>	より大きい
<	より小さい
< >	等しくない
> =	より大きいか等しい
< =	より小さいか等しい

#### 複合検索

複合検索をする場合は検索条件を & (AND) | (OR)で結合します。

#### 比較対象

比較対象で即値を指定する場合、スペースがディリミッタになります。スペースが含まれる即値の場合はシングル・クォート、ダブル・クォート、スラッシュが使えます。

#### 例1

フィールドIDが0の項目が100から1000までの範囲

```
@0 >= 100 & @0 <= 1000
```

#### 例2

フィールド名'会社名'が"TechKnowledge"または"System lab"  
@会社名 = "TechKnowledge" | @会社名 = "System lab"

### 例3

フィールドIDの0と1が同じ値または1000以上  
@0 = @1 | @0 >= 1000

### 例4

フィールドIDが2の項目が"NetScape","Microsoft"以外  
@2 <> 'Microsoft' & @2 <> 'NetScape'

### 注意

このプロパティの各トークンの間はシングル・バイトのスペースで区切ってください。  
複合検索においては、このプロパティの左から右への優先順位で評価されます。括弧を使った優先順位づけなどはサポートされません。理由は、このプロパティの指定どおりにBtrieveの検索パラメータ・ブロックを作成してBtrieveのExtendedオペレーションをしているだけなので、BtrieveのExtendedオペレーションにおいてもパラメータ・ブロックの検索条件に優先順位がない為です。

文字列を検索フィールドに指定する場合英数字に関してDbCompareCaseプロパティを「大文字・小文字の区別なし」に設定することで大文字、小文字を無視した比較が可能です。2バイト文字の英数字にはこのオプションは無効です。

## プロパティ・データ・タイプ

文字列型

## 対象コントロール

拡張リスト、拡張コンボ、グリッド

## DbDelimitChar

---

### 概要

拡張コンボ・ボックスでDbListFieldsプロパティに複数のフィールドを指定する場合、フィールドの間を連結する文字をDbDelimitCharプロパティで指定します。拡張コンボ・ボックスのFormatイベントのパラメータとなる表示文字列もこのプロパティで指定する文字で連結されて渡されます。Formatイベントでフィールドの文字を整形してコンボに表示する場合は、このプロパティにフィールドのデータとして存在しない値を設定して、Formatイベントでフィールドの値を区別します。この文字にはバイナリ・ゼロを指定することはできません。プロパティに何も指定しない場合は、デフォルトとしてスペース文字(0x20,Chr\$(20))をフィールド間の区切り文字として使用します。

Visual Basicのプロパティ・ウィンドウでは文字列データとして存在しないTabのような文字を指定することはできませんから、以下のようにコードで指定することになります。

VbManExComboBox1.DbDelitChar = Chr\$(9)  
VbManExComboBox1.Refresh

### プロパティ・データ・タイプ

文字列型

### 対象コントロール

拡張コンボ

## DbExOperation

---

### 概要

BtrieveのExtendedオペレーションに対応したVbManコントロールで、Extendedオペレーションの種類を指定します。リスト・ボックスやコンボ・ボックスでSortプロパティをTrueに設定した場合はDbListIndexプロパティで設定されるキーに沿ってGet Previous/Next Extendedオペレーションしても、リストに挿入される時にソートされてしまうので、注意してください。

プロパティ値	EXTENDEDオペレーション
0	Get Previous Extended
1	Get Next Extended
2	Step Previous Extended
3	Step Next Extended

### プロパティ・データ・タイプ

整数型

### 対象コントロール

拡張リスト、拡張コンボ、グリッド

## DbField

---

### 概要

VbManコントロールがデータを交換する、テーブルのフィールド名を指定します。プロパティ設定ダイアログでは、DbSource,DbTableプロパティを選択した後でなければ、選択肢は表示されません。

### プロパティ・データ・タイプ

文字列型

### 対象コントロール

ボタン、チェック・ボックス、コンボ、拡張コンボ、拡張リスト、エディット、グリッド、オプション・ボタン、リスト、ピクチャー、スクロール

DbIndex

---

#### 概要

BtrieveのオペレーションでGET PREVIOUS,GET NEXTなどのインデックス・パスを必要とするオペレーションをVBManボタンで実行する場合に指定します。VBManスクロールでは、マウスのクリックなどでGET PREVIOUS,GET NEXTなどのBtrieveオペレーションを発行しますが、このプロパティの指定によって、インデックスのパスを指定します。このプロパティを指定しないと、インデックスのパスはDDF定義の最初のインデックスとしてこれらのオペレーションを実行します。

インデックスの指定はVBMan for Btrieveバージョン1.xではshort型の整数値でしたが、このバージョンではインデックスの名前を指定します。同じフィールド名で異なるインデックスや、セグメント・キーの先頭のフィールド名が別のインデックスでも存在し、複数のフィールド名が選択肢になる場合、DDFビルダーの「インデックス名の追加」で識別可能なインデックス名を設定してください。

### プロパティ・データ・タイプ

文字列型

### 対象コントロール

ボタン、スクロール

DbLeaveResults

---

#### 概要

BtrieveのExtendedオペレーションでデータを取得するVBManコントロールにおいて、検索の結果をメモリに保持したい場合にこのプロパティをTrueに設定します。検索の結果はDbResultIDプロパティに結果IDが返され、この値を元にDbExGetFieldDataメソッド/関数でデータを参照することができます。検索結果が不要になったら、DbExFreeメソッド/関数で結果を開放してください。結果を開放しないで、プログラムを終了した場合は動作を保証できません。

---

8

Scalable SQLのマニュアルではNamed Indexと記述されます。Create IndexというSQL文で指定されるインデックスの名前です。

## プロパティ・データ・タイプ

論理型

### 対象コントロール

拡張リスト、拡張コンボ、グリッド

#### DbListFields

---

##### 概要

拡張リスト等複数のフィールドをリストするコントロールにおいて表示するフィールドを指定します。フィールドはカンマ区切りで複数を指定できます。このプロパティがプロパティ・ダイアログで指定可能になるためには、DbListSource、DbListTableプロパティをこのプロパティより先に指定していることが必要です。

バージョン1.xではカンマ区切りで整数値のフィールドIDを指定していましたが、このバージョンでは、カンマ区切りでフィールドの名前を指定します。

カンマはシングル・バイトのカンマを指定することに注意してください。

以下はVisual Basicにおけるサンプル・コードです。

```
Sub Command1_Click()  
VBManExListBox1.DbListFields = "コード,商品名,仕入先コード"  
VBManExListBox1.Refresh  
End Sub
```

## プロパティ・データ・タイプ

文字列型

### 対象コントロール

拡張リスト、拡張コンボ、グリッド

#### DbListIndex

---

##### 概要

BtrieveデータをリストするVBManコントロールでは、GET FIRST、GET NEXTの順または、GET FIRST、GET NEXT EXTENDEDなどの順でBtrieveオペレーションを発行します。これらのオペレーションはこのDbListIndexプロパティで指定されたインデックス順にデータを読み込みます。

DbListIndexプロパティであるインデックス順にデータを読み込んだ場合、対象となるコントロールのSortプロパティをTrueに設定すると、コントロールにデータが挿入される時点でソートされてしまい、このインデックス指定順にならない場合がありますのでご注意ください。



プロパティは文字型で、コントロールのプロパティ・ダイアログで指定する場合は、DbListSource,DbListTableプロパティが設定されていないと、選択肢がリストされないのでプロパティの設定順にご注意ください。

### **プロパティ・データ・タイプ**

文字列型

### **対象コントロール**

ボタン、スクロール

## DbListSource

---

### **概要**

リスト系のVBManコントロールが参照するVBManデータベース・コントロールのNameプロパティの値を指定します。このプロパティを設定する対象となるVBManコントロールより先にフォームにVBManデータベース・コントロールが置かれていない場合には、プロパティ設定ダイアログで選択肢が表示されません。実行時にはこのプロパティで指定されるVBManコントロールのDDFDirプロパティで指定されるテーブル、フィールド情報にしたがって、Btrieveデータをリストします。

### **プロパティ・データ・タイプ**

文字列型

### **対象コントロール**

リスト、コンボ、拡張コンボ、拡張リスト、グリッド

## DbListTable

---

### **概要**

リスト系のVBManコントロールがデータをリストするテーブル名を指定します。プロパティ設定ダイアログでは、DbListSourceが先に設定されていないと、選択肢が表示されません。

### **プロパティ・データ・タイプ**

文字列型

### **対象コントロール**

リスト、コンボ、拡張コンボ、拡張リスト、グリッド

## DbLocks

---

## 概要

DbLocksプロパティはBtrieveデータベースに対するオペレーション実行時にかけるロックのタイプを指定します。シングル、マルチ、Wait, NoWaitを選択します。

プロパティ値	意味
0	ロックしない
1	シングル・ウェイト
2	シングル・ノーウェイト
3	マルチ・ウェイト
4	マルチ・ノーウェイト

10

ロックを解除する場合はDbUnlockメソッドまたは関数を使用することでも可能です。

## プロパティ・データ・タイプ

整数型

## 対象コントロール

ボタン、拡張コンボ、拡張リスト、グリッド

### DbMaxRecords

---

## 概要

BtrieveのExtended系オペレーションでデータを取得するVBManコントロールで、検索結果の最大レコード件数を設定します。Btrieveでは結果はリニアなメモリに返されるため、65,536バイト以上を越えることができません。65,536をDbListFieldsプロパティで指定する抽出するフィールドの合計サイズで割った値以上は設定できません。

## プロパティ・データ・タイプ

整数型

## 対象コントロール

拡張コンボ、拡張リスト、グリッド

9

DbAccessメソッド・関数でロックをかける場合は第一パラメータの値にロック・バイアス値を加算します。

10

Singleロックなどの場合は別のレコードにロックするとそれ以前のロックは解除されます。

**概要**

VBManボタンがクリックされたときに発行するBtrieveオペレーションを指定します。以下の値が指定可能です。

プロパティ値	BTRIEVEオペレーション
0	Insert
1	Update
2	Delete
3	GetEqual
4	GetFirst
5	GetLast
6	GetNext
7	GetPrevious
8	GetGreater
9	GetLessThan
10	GetGreaterOrEqual
11	GetLessThanOrEqual
12	StepFirst
13	StepLast
14	StepNext
15	StepPrevious

**プロパティ・データ・タイプ**

整数型

**対象コントロール**

ボタン

**概要**

AllowDelete/AllowUpdateをTrueに設定したグリッドコントロールに表示されているレコードのBtrieveファイル上の物理位置を保持します。物理位置を取得する場合にはゼロ・オフセットのインデックスを指定して4バイトの物理位置を得ることができます。Fixedセルにはデータは表示されませんから、グリッドに表示されている先頭のレコードの物理位置を得る場合には値0をプロパティ・インデックスとして指定します。取得した物理位置はDbGetDirectメソッドに指定してレコード全体を取得する等の利用が可能です。

### プロパティ・データ・タイプ

長整数型

### 対象コントロール

グリッド

### サンプル・コード

Dim IPos As Long

Dim rc As Integer

IPos = VBManGrid1.DbPhysicalPosition(3)

rc = VBManDb1.DbGetDirect("TEST\_TABLE",IPos,"INDEX-1")

DbResultID

---

### 概要

BtrieveのExtended系オペレーションでデータをリストするVBManコントロールのDbLeaveResultsプロパティをTrueに設定した場合には、このプロパティに結果IDが返されます。このIDを基にDbExGetFieldDataメソッド/関数で検索結果を得ることができます。検索結果が不要になったら、このIDをDbExFreeメソッド/関数で開放してください。プログラムの終了時に開放されていない場合の動作については保証できません。

### プロパティ・データ・タイプ

文字列型

### 対象コントロール

拡張コンボ、拡張リスト、グリッド

DbSavePosition

---

### 概要

このプロパティをTrueに設定すると、検索結果のBtrieveレコード物理位置をItemDataプロパティに保持します。この物理位置からレコード全体を取得するにはDbGetDirectメソッドまたは

バージョン1.xコンパチブル・関数を使います。以下はコード例です。

```
Dim Position As Long
```

```
Dim rc As Integer
```

‘ リストの先頭にあるレコードをカレント・レコードにする。

```
Position = VBManComboBox1.ItemData(0)
```

```
rc = VBMan1.DbGetDirect("CUSTOMER",Position,"INDEX1")
```

```
If rc <> 0 Then
```

```
    MsgBox "Btrieve error " & CStr(rc)
```

```
    Stop
```

```
End If
```

### プロパティ・データ・タイプ

論理型

### 対象コントロール

拡張リスト・ボックス、拡張コンボ・ボックス

#### DbSkipRecords

---

### 概要

BtrieveのExtended系オペレーションでデータを取得するVBManコントロールで、DbConditionsプロパティで指定される検索条件に合致しないレコード件数を指定します。このプロパティの値だけ検索結果に合致しないレコードを読み込んだ場合、Btrieveはステータス 64<sup>11</sup> を返します。このステータスを得た場合はこの値を大きくしてください。0を指定した場合はシステムで定義された最大値となります。Btrieve 6.15ではこの値は4,095です。

### プロパティ・データ・タイプ

整数型

### 対象コントロール

拡張コンボ、拡張リスト、グリッド

#### DbSource

---

---

<sup>11</sup> Filter Limit Reached

## 概要

VBManコントロールがデータを交換するVBManデータベース・コントロールのNameプロパティの値を指定します。このプロパティを設定する対象となるVBManコントロールより先にフォームにVBManデータベース・コントロールが置かれていない場合には、プロパティ設定ダイアログで選択肢が表示されません。

## プロパティ・データ・タイプ

文字列型

## 対象コントロール

ボタン、チェック・ボックス、コンボ、拡張コンボ、拡張リスト、エディット、グリッド、オプション・ボタン、リスト、ピクチャー、スクロール

DbTable

---

## 概要

コントロールがデータを交換するテーブル名を指定します。プロパティの設定ダイアログではDbSourceプロパティを選択した後でなければ、選択肢は表示されません。

## プロパティ・データ・タイプ

文字列型

## 対象コントロール

ボタン、チェック・ボックス、コンボ、拡張コンボ、拡張リスト、エディット、グリッド、オプション・ボタン、リスト、ピクチャー、スクロール

DbTabStops

---

## 概要

拡張リスト・ボックスにおいて複数のフィールドを表示するようにDbListFieldsプロパティで指定した時、それぞれのフィールドの表示開始カラム位置をこのプロパティで設定します。タブ位置はカンマで区切り、表示するフィールドの数だけ指定します。

## プロパティ・データ・タイプ

文字列型

## 対象コントロール

拡張リスト

## Visual Basicサンプル

VBManExListBox1.DbTabStops = "1,20,30"

VBManExListBox1.Refresh

## DbUpdateOption

---

### 概要

リスト・ボックスやコンボ・ボックスにデータを読み込むタイミングを設定します。設定値は以下です。

プロパティ値	動作
0	データはリストされません。Refreshメソッドを発行してデータをリストしてください。
1	初期化時。コントロールがロードされた時、またはこのコントロールが参照しているデータベース・コントロールがロードされた時にデータをリストします。
2	フォーカス時にデータをリストします。大量のデータをリストする場合、フォーカス時に設定すると、フィールドを移動するときにパフォーマンスに問題がでる場合が考えられます。データ量をよく見積もってからこの値を設定してください。

### プロパティ・データ・タイプ

整数型

### 対象コントロール

コンボ、リスト、拡張リスト、拡張コンボ

## DbValueFalse

---

### 概要

Btrieveデータ・ベースのフィールドの値がこのプロパティで指定される値と同一になった場合に、チェック・ボックスの場合は非チェック状態に、オプション・ボタンの場合は非選択状態になります。VBManバージョン1.xでは論理値しか設定できませんでしたが、VBMan Controls for Btrieveからは、フィールドの値を表示形式にした値とこのプロパティに設定した値を比較するように変更されました。

### プロパティ・データ・タイプ

文字列型

## 対象コントロール

チェック・ボックス、オプション・ボタン

### DbValueIndeterminate

---

#### 概要

Btrieveデータ・ベースのフィールドの値がこのプロパティで指定される値と同一になった場合に、チェック・ボックスはグレー状態<sup>12</sup> になります。

#### プロパティ・データ・タイプ

文字列型

## 対象コントロール

チェック・ボックス

### DbValueTrue

---

#### 概要

Btrieveデータ・ベースのフィールドの値がこのプロパティで指定される値と同一になった場合に、チェック・ボックスの場合はチェック状態に、オプション・ボタンの場合は選択状態になります。VBManバージョン1.xでは論理値しか設定できませんでしたが、VBMan Controls for Btrieveからは、フィールドの値を表示形式<sup>13</sup> にした値とこのプロパティに設定した値を比較するように変更されました。

#### プロパティ・データ・タイプ

文字列型

## 対象コントロール

チェック・ボックス、オプション・ボタン

### DDFDir

---

---

<sup>12</sup> Win32 APIリファレンスのBM\_SETCHECKメッセージの説明を参照してください。WPARAMの値が2の場合にこの状態になります。

<sup>13</sup> たとえば、バイナリの整数型をプリントできる文字(printable character)に変換すること。



## 概要

VBManデータベース・コントロールのプロパティでDDFファイルが存在するディレクトリを指定します。指定するディレクトリには以下の3ファイルが存在していることが必要です。

FILE.DDF  
FIELD.DDF  
INDEX.DDF

## プロパティ・データ・タイプ

文字列型

## 対象コントロール

データベース・コントロール

DistName

---

## 概要

このプロパティはDDFDirプロパティをアプリケーション配布時に置き換えるVBMAN.INIファイルのセクション名を指定します。詳細はVBMan Controls for Btrieveで作成したアプリケーションの配布の章を参照してください。

## プロパティ・データ・タイプ

文字列型

## 対象コントロール

データベース・コントロール

Format

---

## 概要

FormatプロパティはVisual Basicを開発言語としている場合のみ、設定可能なプロパティです。このプロパティの書式はVisual BasicのFormat\$関数の整形文字列が指定可能です。たとえば、数値型であれば、このプロパティには“###,###,###”のような文字列を指定すると、数値入力された場合に、以下で説明するFormatOption設定のタイミングで”123,456”のように文字列が整形され、VBManエディットに設定・表示されます。

VBManエディットは関連するフィールドのデータ型がDate型、Time型の場合、このFormatプロパティに何も指定されていなければ、それぞれ”YYYY/MM/DD”, ”HH:MM:SS”の形式でデータを整形します。

## プロパティ・データ・タイプ

文字列型

## 対象コントロール

エディット

### 注意

リスト・ボックス、コンボ・ボックス、拡張リスト・ボックス、拡張コンボ・ボックス、グリッドにはFormatイベントが存在し、エディットにはFormatプロパティが存在します。

## FormatOption

---

### 概要

Formatプロパティを設定したときに、VBManエディットの動作オプションを設定します。以下の値がプロパティに設定可能です。

プロパティ値	オプション動作
0	オプションを設定しません。
1	文字単位でフォーマット。キーボードからの文字入力が発生する度にFormatプロパティの指定があれば、これにしたがって整形した文字列をVBManエディットに設定します。
2	右揃え。Format文字列で整形した文字列をMaxLengthプロパティにしたがって、右揃えで表示します。また、VBManエディットがフォーカスを受け取った場合、文字列の右にカーソルを設定します。かならずMaxLengthプロパティに適切な値を設定することが必要なことに注意してください。
3	右そろえ+文字単位フォーマット。このプロパティの値1と2の動作をあわせておこないます。

## プロパティ・データ・タイプ

整数型

## 対象コントロール

エディット

## FormatString

---

## 概要

グリッドのカラム単位に整形文字列を指定します。整形文字列の形式はVisual BasicのFormat関数とコンパチブルです。ホスト言語がVisual Basicの場合のみこのプロパティの動作が保証されます。カラムの指定はプロパティのインデックスで0ベースになります。

## プロパティ・データ・タイプ

文字列型配列

## 対象コントロール

グリッド

## サンプル・コード

```
With VBMGrid1
    .FormatString(0) = "@@@@@@@@@@"
    .FormatString(1) = "###,###,###"
    .Refresh
End With
```

---

## NumericMask

## 概要

VBManエディット<sup>14</sup>コントロールに入力される文字を数値と特定の文字に限定します。マスクを通過する文字は数値 と,.-+です。Formatプロパティに###,###,###のような数値を整形する文字列を設定する場合、このプロパティをTrueに設定すると、ユーザーが誤って、英文字を入力したりすることを防止できます。

## プロパティ・データ・タイプ

論理型

## 対象コントロール

エディット

---

## OpenMode

## 概要

---

<sup>14</sup> シングル・バイトのみ。

VBManデータベース・コントロールに関連するBtrieveファイルのオープン・モード<sup>15</sup>を指定します。オープン・モードとして指定可能なプロパティ値は以下です。

プロパティ値	オープン・モード
0	ノーマル
1	アクセラレーテッド
2	リード・オンリー
3	ベリファイ
4	エクスクルーシブ
5	シングル・エンジン・ファイル・シェア
6	マルチ・エンジン・ファイル・シェア

### プロパティ・データ・タイプ

整数型

### 対象コントロール

データベース・コントロール

OwnerName

---

### 概要

VBManデータベース・コントロールに関連するBtrieveファイルのオーナー名を指定します。

### プロパティ・データ・タイプ

文字列型

### 対象コントロール

データベース・コントロール

ReadOnly

---

### 概要

このプロパティをTrueに設定すると、VBManエディット・コントロールを読み取り専用を設定します。キーボードによる、文字は受け付けなくなります。

---

<sup>15</sup> Btrieveファイルのオープン・モードの詳細はBtrieveプログラマーズ・マニュアルのOpenオペレーションにあります。

## プロパティ・データ・タイプ

論理型

## 対象コントロール

エディット

SetColNames

---

### 概要

このプロパティの値をTrueに設定するとグリッドにデータを表示するときに、最初のFixedカラムにDbListFieldsプロパティで指定されたフィールド名を設定、表示します。グリッドのFixedColsプロパティの値は1に設定してください。

## プロパティ・データ・タイプ

論理型

## 対象コントロール

グリッド

SetColWidth

---

### 概要

SetColNamesプロパティとこのプロパティに値Trueを設定した場合はグリッドにデータ・ベースから読み、カラム名をFixedセルに設定する時にセルの幅を自動的に調整します。設定されるセルの幅はフィールド名の文字列がすべて表示される幅となります。

## プロパティ・データ・タイプ

論理型

## 対象コントロール

グリッド

SetRecordNumber

---

### 概要

このプロパティの値をTrueに設定するとグリッドにデータを表示するときに、最初のFixedローにレコード番号を設定、表示します。グリッドのFixedRowsプロパティの値は1に設定してください。

## プロパティ・データ・タイプ

論理型

## 対象コントロール

グリッド

### TextAlign

---

#### 概要

グリッドのセルに表示される文字列のアライメントを指定します。セルの指定は列単位での指定が可能です。列の指定はプロパティの配列で指定します。固定セルに関してはAlignFixedCellsプロパティをFalseに設定すると右つめの指定は無効となります。

指定可能なプロパティ値は以下です。

プロパティ値	アライメント
0	左つめ
1	右つめ

## プロパティ・データ・タイプ

整数型

## 対象コントロール

グリッド

#### サンプル・コード

```
Private Sub Form_Load
With VBManGrid1
.TextAlign(1) = 1           '右つめ
.ColWidth(1) = 1000       'セル幅を設定
.ImeMode(1) = 1           'IMEモードを設定
.CellMaxLength(1) = 10    '入力の最大長を設定
.Col = 1
.Row = 0
.Text = "売上高"
.Refresh                   'データを読み込む
End With
End Sub
```

### 概要

このプロパティをTrueに設定すると、VBManエディット・コントロールに英数文字を入力された場合にはアップパーケースに変換します。

### プロパティ・データ・タイプ

論理型

### 対象コントロール

エディット

## カスタム・イベント・リファレンス

---

VbMan Controls for Btrieveのカスタム・イベントについて解説します。

### AfterColUpdate

---

#### 書式

Object\_AfterColUpdate()

#### パラメータ

なし

#### 概要

グリッドコントロールではAllowUpdateプロパティをTrueに設定した場合にセルが編集可能になります。編集可能となったセルからフォーカスが右矢印キーまたはEnterキーにて次のセルに移動するときに編集内容がTextプロパティとして保持された後に当イベントが発生します。

#### 対象コントロール

グリッド

### AfterDelete

---

#### 書式

Object\_AfterDelete()

#### パラメータ

なし

#### 概要

AllowDeleteプロパティをTrueに設定したグリッドでレコード削除が実行された後に発生するカスタム・イベントです。

#### 対象コントロール

グリッド

### AfterUpdate

---

#### 書式



Object\_AfterUpdate()

### パラメータ

なし

### 概要

AllowUpdateプロパティをTrueに設定したグリッド・コントロールでレコードのUpdate終了後に発生するイベントです。

### 対象コントロール

グリッド

---

## BeforeColUpdate

### 書式

Object\_BeforeColUpdate( Cancel As Integer, ColValue As String )

### パラメータ

Cancel	0以外の値をイベント無いで設定した場合にはカラムの変更をキャンセルします。
ColValue	編集されたテキストの値です。編集前のテキストはTextプロパティに設定されています。

### 概要

AllowUpdateプロパティをTrueに設定したグリッド・コントロールではカラムの編集が可能になります。当イベントはカラムが編集されて変更がグリッドのTextプロパティとして設定される前に発生します。アプリケーションはカラムの値を検査して不正と判断した場合にはCancelパラメータに0以外の値をセットすることで編集以前の値に戻すことができます。

### 対象コントロール

グリッド

---

## BeforeDelete

### 書式

Object\_BeforeDelete( Cancel As Integer )

### パラメータ

Cancel	0以外の値をイベント無いで設定した場合にはレコードのDeleteをキャンセルします。
--------	--

## 概要

### 対象コントロール

グリッド

BeforeUpdate

---

## 書式

Object\_BeforeUpdate( Cancel As Integer )

### パラメータ

Cancel	0以外の値をイベント無いで設定した場合にはレコードのUpdateをキャンセルします。
--------	--

## 概要

AllowUpdateプロパティをTrueに設定したグリッド・コントロールでレコードのUpdateを実行する直前に発生するイベントです。アプリケーションはイベント内では変更されるデータの妥当性を検証して、データに不正が発見された場合はエラー・メッセージを表示してCancelパラメータに0以外の値を設定することでレコードのUpdate処理を無効にできます。

### 対象コントロール

グリッド

BtrieveError

---

## 書式

Object\_BtrieveError( TableName As String,  
Status As Integer )

### パラメータ

TableName	Btrieve呼び出しの対象となったテーブル名。
Status	Btrieveからのステータス。

## 概要

VbManカスタム・コントロールがBtrieveを呼び出した結果、ステータスが0以外の値がBtrieveから返された場合にこのイベントが発生します。

VbManボタンなどのオペレーション時にBtrieveからエラーが返されると、このイベントが発生した後はデフォルトでメッセージ・ボックスが表示されます。このメッセージ・ボックスを表示したくない場合は、第2パラメータの値に0をセットします。以下はVisual Basicでのコード例です。

```
Sub BtrieveError( TableName As String, Status As Integer )
MsgBox "Btrieveエラーです " & CStr(Status)
Status = 0
End Sub
```

## 対象コントロール

VbManカスタム・コントロール

Format

---

## 書式

Object\_Format( LineData As String )

## パラメータ

LineData	リスト・ボックス/コンボ・ボックスに表示する文字列
----------	---------------------------

## 概要

リスト・ボックス等にデータを挿入する直前にこのイベントが発生します。

リスト・ボックス、コンボ・ボックスではこのイベントが発生した時にはパラメータとして渡されたフィールドを含むデータのカレント・レコードが存在します。したがってDbGetFieldDataメソッドまたは関数を呼び出すことで、イベント・パラメータLineData以外のフィールドの値を得ることができます。

拡張リスト、拡張コンボ、グリッドではFormatイベントが発生したときには、BtrieveのExtended系のオペレーションが終了しているので、カレント・レコードは最後に取得したレコードになります。したがって、FormatイベントでDbGetFieldDataメソッドを使っても常に同じレコードのフィールドの値が返されるので、このメソッドを使う意味はありません。これらのコントロールではDbListFieldsプロパティに複数のフィールドが指定可能でそれらの値がLineDataプロパティで渡されるので、他のフィールドを参照するのではなく、DbListFieldsプロパティに参照したいフィールドを記述することになります。これは同じテーブルに関することで、他のテーブルを参照する場合は、LineDataに他のテーブルのキーとなる値が存在すれば、DbAccessメ

ソッドを使って他のテーブルの値をGET EQUALオペレーション等で参照することは可能です。<sup>16</sup>

拡張リストではTabStopsプロパティを指定した場合、フィールドの間はタブ文字 で連結されてFormatイベントに渡されます。

拡張コンボではDbDelimitCharプロパティの値で指定した文字がフィールドの間を連結してFormatイベントに渡されます。

## 対象コントロール

コンボ、リスト、拡張リスト、拡張コンボ、グリッド

### SetData

---

#### 書式

Object\_SetData()

#### パラメータ

無し

#### 概要

VBManボタンコントロールにおいてDbOperationプロパティにデータを登録するオペレーションがセットされている場合、ユーザーがボタンを押したときにこのイベントが発生します。このイベントではDbSetFieldDataメソッド/関数を使って、フォームのコントロールにないテーブルのフィールドにデータをセットしたり、フォームのコントロールへの入力の正当性をチェックして不正であれば、メッセージを表示して、AbortプロパティをTrueにセットすれば、ボタンを押したオペレーションを中断することもできます。イベントのタイミングはフォームにある関連するコントロールのデータをVBManデータ・ベース・コントロールのレコード・イメージのバッファに転送した後で、Btrieveを呼び出す直前です。以下はVisual Basicによるサンプル・コードです。

‘ ボタンのNameプロパティはInsertDataとなっている例です。

‘ このボタンのDbSourceプロパティはVBMan1となっています。

```
Sub InsertData_SetData()  
If Val( ItemPrice.Text ) <= 0 Then  
    MsgBox “商品の値段が入力されていません”  
    InsertData.Abort = True  
    Exit Sub  
End If  
‘ 在庫個数を0にセット  
VBMan1.DbSetFieldData(“商品”,“在庫”,“0”)  
End Sub
```

---

<sup>16</sup> アスキー・コード9

VBManボタンのイベントとデータベースへのアクセスに関する順序を以下に示します。

**対象コントロール**  
ボタン

## カスタム・メソッド・リファレンス

---

コンパチビリティの章でも説明しましたが、VBMan Controls for Btrieveでは、従来のバージョン1.xコンパチブル関数に加えて、VBManデータベース・コントロールのメソッドとして、バージョン1.x関数相当のメソッドがデータ・ベース・コントロールに追加されました。

Visual Basic 32bit版では関数からメソッドへの変更に伴って、関数宣言は必要なくなりましたが、メソッドのパラメータでの定数はBtrieve.basファイルに定義されていますのでこのファイルをプロジェクトに追加してメソッドをご利用ください。

### DbAbortTransaction

---

Object.DbAbortTransaction() As Integer

#### 概要

トランザクションの中止を宣言します。

#### パラメータ

なし。

#### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外の正の値Btrieveのステータス・コードが返されます。

### DbAccess

---

Object.DbAccess(BtrieveOpCode As Integer,  
                  TableName As String,  
                  KeyName As String) As Integer

#### 概要

BtrieveOpCodeで指定されるBtrieveの機能呼び出します。

#### パラメータ

##### BtrieveOpCode

Btrieveの機能番号を指定します。VBManに添付されるBTRIEVE.BASファイルに定義されています。このメソッドで使用できるBtrieveオペレーション・コードはこのファイルに定義されているものに限定されます。

### TableName

Btrieveオペレーションを発行するテーブルの名前を指定します。

### KeyName

Btrieveオペレーションに関連するキーの名前を指定します。テーブル定義上、同じキーの名前が存在する場合はVBMan DDFビルダーでキーに別名を設定してください。

### **戻り値**

正常終了ならば0が返ります。負の値はVBManエラー・コード一覧を参照してください。それ以外の正の値Btrieveのステータス・コードです。

### **Visual Basicサンプル**

```
Dim rc%
With VBManDb1.
  rc% = .DbSetFieldData("従業員","社員番号","066217")
  rc% = .DbAccess(BTR_GET_EQUAL,"従業員","社員キー")
  If rc% <> 0 Then
    MsgBox "Btrieveの呼び出しに失敗しました" + Str$(rc%)
    Exit Sub
  End If
  '読み込んだ結果をフォーム上のコントロールに転送
  rc% = .DbTransferToControl("従業員")
End With
```

### DbAccessByID

---

```
Object.DbAccessByID(BtrieveOpCode As Integer,
                    TableID As Integer,
                    KeyID As Integer) As Integer
```

### **概要**

BtrieveOpCodeで指定されるBtrieveの機能呼び出します。テーブルとキーは整数値のIDを指定します。IDによるテーブル、フィールドの指定はVBManバージョン1.xでの仕様と同様になりますので既存のバージョン1.xコードを移植する際に有効です。

### **パラメータ**

#### BtrieveOpCode

Btrieveの機能番号を指定します。VBManに添付されるBTRIEVE.BASファイルに定義されてい

ます。このメソッドで使用できるBtrieveオペレーション・コードはこのファイルに定義されているものに限定されます。

#### TableID

Btrieveオペレーションを発行するテーブルIDを指定します。

#### KeyID

Btrieveオペレーションに関連するキーIDを指定します。キーIDはDDFビルダーで定義されたキーを識別する整数値です。ゼロ・オフセットで指定します。

#### **戻り値**

正常終了ならば0が返ります。負の値はVBManエラー・コード一覧を参照してください。それ以外の正の値Btrieveのステータス・コードです。

#### **サンプル・コード**

```
Const TID_EMP = 10
Const FID_EMP_NO = 1
Const KID_EMP_NO = 0

Dim rc%
With VBManDb1.
    rc% = .DbSetFieldDataByID(TID_EMP,FID_EMP_NO,"066217")
    rc% = .DbAccessByID(BTR_GET_EQUAL,TID_EMP_NO,KID_EMP_NO)
    If rc% <> 0 Then
        MsgBox "Btrieveの呼び出しに失敗しました" + Str$(rc%)
        Exit Sub
    End If
End With
```



## DbBeginConCurTransaction

---

Object.DbBeginConCurTransaction(LockBias As Integer) As Integer

### 概要

コンカレント・トランザクションの開始を宣言します。

### パラメータ

#### LockBias

トランザクション・ロックを指定。値は100,200,300,400,500を指定可能。詳細はBtrieveプログラマーズ・マニュアルを参照してください。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

## DbBeginTransaction

---

Object.DbBeginTransaction(LockBias As Integer) As Integer

### 概要

トランザクションの開始を宣言します。

### パラメータ

#### LockBias

トランザクション・ロックを指定。値は100,200,300,400を指定可能。詳細はBtrieveプログラマーズ・マニュアルを参照してください。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

## DbClearControlData

---

Object.DbClearControlData(TableName As String) As Integer

### 概要

アクティブなフォームにあるコントロールのデータ<sup>1/</sup>を初期化します。Btrieveのデータ入出力に使われるメモリ・バッファは初期化されません。こちらを初期化したい場合は、DbClearFieldBufferメソッドを使ってください。

## パラメータ

TableName

データを初期化したいコントロールが関連しているテーブル名

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

## DbClearFieldBuffer

---

Object.DbClearFieldBuffer( TableName As String ) As Integer

## 概要

指定したテーブルのデータ・バッファを初期化します。データ・バッファはBtrieveとのデータを交換するメモリ・エリアです。

## パラメータ

TableName

テーブルの名前を指定します。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

## DbClose

---

Object.DbClose( TableName As String ) As Integer

## 概要

指定されるテーブルに関連するBtrieveファイルをクローズします。このメソッドを呼び出す以前にBtrieveファイルはオープンされている必要があります。

---

<sup>1/</sup> VBManエディットのテキスト・プロパティを空にします。

## パラメータ

TableName

テーブルの名前を指定します。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

---

DbCloseAll

Object.DbCloseAll() As Integer

## 概要

DDFに定義されたBtrieveファイルをすべてクローズします。

## パラメータ

なし

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

---

DbCreate

Object.DbCreate( TableName As String ) As Integer

## 概要

TableNameで指定されるテーブルに関連するBtrieveファイルを生成(Create)します。レコード長、インデックスの構成などはDDFの定義を参照します。このメソッドを呼び出す時には関連するBtrieveファイルはクローズされていることが必要です。サーバーにあるBtrieveファイルをマルチ・ユーザーで使用する場合は一つのクライアントからDbCloseしても、他でオープンしていれば、このメソッドは成功しません。すでにBtrieveファイルが存在するような場合は上書きされますので注意してください。生成されるBtrieveファイルのページ・サイズは4,096となります。

## パラメータ

TableName

テーブルの名前を指定します。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

## Visual Basicサンプル

```
Dim rc As Integer
rc = VBMan1.DbClose("月次ファイル")
Kill "c:\data\月次.btr"
rc = VBMan1.DbCreate("月次ファイル")
If rc <> 0 Then
    MsgBox "Btrieve createステータス " & CStr(rc)
    Stop
End If
rc = VBMan1.DbOpen("月次ファイル")
If rc <> 0 Then
    MsgBox "Btrieve openステータス " & CStr(rc)
    Stop
End If
```

## DbEndTransaction

---

Object.DbEndTransaction() As Integer

## 概要

トランザクションの終了を宣言します。

## パラメータ

なし。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

## DbExAccess

---

```
Object.DbExAccess( OpCode As Integer,
    TableName As String,
    SearchCondition As String,
    Fields As String,
    IndexName As String,
    SkipRecords As Integer,
```

MaxRecords As Integer,  
IgnoreCase As Integer,  
ResultID As Integer,  
NumberOfResults As Integer ) As Integer

## 概要

BtrieveのExtended関数を実行します。

## パラメータ

### TableName

テーブル名を指定します。

### OpCode

Extendedオペレーション・コードを指定します。指定できる値はBTRIEVE.BASに宣言してある以下の値です。

BTR_GET_PREVIOUS_EXTENDED	37
BTR_GET_NEXT_EXTENDED	36
BTR_STEP_PREVIOUS_EXTENDED	39
BTR_STEP_NEXT_EXTENDED	38

## SearchCond

検索条件文字列を指定します。拡張リスト・ボックス、拡張コンボ・ボックスの検索条件文字列と同じ形式で指定します。

フィールドは@をプリフィックスとして比較式の左辺に指定します。@プリフィックスの後にはフィールド名を記述します。バージョン1.xとのコンパチビリティのために、フィールドIDも指定できます。（例；フィールドIDの10は@10と指定する）比較演算子は以下を指定します。

=	等しい
>	より大きい
<	より小さい
<>	等しくない
>=	より大きいか等しい
<=	より小さいか等しい

複合検索をする場合は検索条件を& (AND)または| (OR)で結合します。比較対象を即値で指定する場合はスペースがディリミッタになります。スペースが含まれる即値を指定する場合はシングル・クォート、ダブル・クォート、スラッシュで文字列を囲みます。以下に検索文字列の例を示します。

@0 >= 100 & @1 <= 1000  
@製品 = "VBMan" | @製品 = "Visual Basic"  
@2 <> /Microsoft/ & @2 <> /Borland/

### Fields

検索結果として抽出するフィールドを指定します。フィールド名またはフィールドIDをカンマで区切ったものを指定します。フィールドはIDをカンマで区切った文字列を指定します。フィールドIDはDDFビルダーの表示値を0オフセットにした値です。(例: 10,22,34)

### IndexName

GET NEXT/PREVIOUS EXTENDEDオペレーションを実行する際のインデックスのパスを指定します。DbAccess関数の第三パラメータと同じ形式で指定します。カレント・レコードが確立されていない場合、GET NEXT/PREVIOUS EXTENDEDオペレーションはエラーを返します。

### SkipRecords

検索条件を満たさない場合にスキップするレコード数を指定します。

### MaxRecords

検索結果の最大レコード件数を指定します。Btrieveでは結果が65,536バイトをこえることができません。65,535をFieldsで指定した抽出するフィールドのデータ長の合計値で割った数以上は設定できません。

### IgnoreCase

文字列フィールドを比較する場合、0以外の値を指定すると英数字に関して大文字、小文字の区別無く比較を実行します。2バイト文字にはこのオプションはサポートされません。

### ResultID

結果IDが整数値で返されます。結果を取得するにはこのIDを指定してDbExGetFieldData,DbExGetPosition関数を使用します。結果が不要になったらこのIDでDbExFree関数を呼び出し資源を解放します。

### NumberOfResults

検索結果件数が返されます。

## **戻り値**

正常ならば0が返されます。正の整数値が返された場合はBtrieveからのエラー・コードです。負の整数値はVBManエラー・コード一覧を参照してください。

また、エラー9,22,60,64,65等が返った場合でも結果が存在する場合もあることに注意してください。

以下にこのオペレーションでBtrieveから返されるエラー・コードを抜粋します。

03	FILE NOT OPEN
06	Invalid Key Number
07	Different Key Number
08	Invalid Position
09	End Of File
22	Data Buffer Length
60	Reject Count Reached
61	Work Space Too Small
62	Incorrect Descriptor
6	Filter Limit Reached
4	
65	Incorrect Field Offset
82	Lost Position

### 注意

カレント・レコードを検索の対象から除外したい場合はDbExSetModeメソッドで指定します。

### Visual Basicサンプル

```
Dim rc As Integer
```

```
Dim ResultID As Integer
```

```
Dim NumOfRecords As Integer
```

```
Dim Fields As String
```

```
Dim SearchCond As String
```

```
Dim a$,l As Integer
```

```
SearchCond = "@売値 > 100 & @売値 < 200"
```

```
Fields = "コード,売値,仕切値,仕入先"
```

```
rc = DbExAccess( BTR_GET_NEXT_EXTENDED, 0, SearchCond, Fields,  
0,127,100,0,ResultID,NumOfRecords )
```

```
If rc <> 9 & rc <> 0 & rc <> 60 Then
```

```
    MsgBox "Btrieve Extended Opeartion Error " & Str$(rc)
```

```
    Stop
```

```
End If
```

```
' 結果をリスト・ボックスに追加
```

```
For l = 0 To NumOfRecords -1
```

```
    a$ = DbExGetFieldData(ResultID,0,l)
```

```
    a$ = a$ & " " & DbExGetFieldData(ResultID,1,l)
```

```
    ResultListBox.AddItem a$
```

Next |

‘結果を保持するメモリを解放

```
rc = DbExFree(ResultID)
```

---

## DbExFree

---

```
Object.DbExFree( ResultID As Integer )
```

### 概要

DbExAccess関数によって返される結果に付随する資源を解放します。

### パラメータ

ResultID

DbExAccess関数の結果として返される結果IDを指定します。

### 戻り値

なし。

### 注意

DbExAccess関数の結果は検索結果件数と抽出するフィールドのサイズと件数に比例してメモリを消費します。不要になった結果はこの関数を呼び出して解放してください。

---

## DbExGetFieldData

---

```
Object.DbExGetFieldData( ResultID As Integer,  
                          FieldID As String,  
                          Location As Integer) As String
```

### 概要

DbExAccessメソッドの結果をフィールド単位で取得します。

### パラメータ

ResultID

DbExAccessメソッドの結果として返される結果IDを指定します。

FieldID

フィールド名を文字列で指定します。

Location

DbExAccessは検索結果として複数のレコード・イメージを返します。Locationパラメータは所得する結果の位置を指定します。0オフセットで結果位置を整数で指定します。DbExAccessで



は検索結果件数を返しますがその値以上を指定することはできません。

### 戻り値

パラメータ不正、メモリが不足してVisual Basic文字列が作成できない場合にヌル文字列を返します。正常の場合はフィールドの値を文字列で返します。

## DbExGetPosition

---

Object.DbExGetPosition( ResultID As Integer,  
Location As Integer ) As Long

### 概要

DbExAccess関数の結果を物理レコード位置単位で所得します。

### パラメータ

#### ResultID

DbExAccess関数の結果として返される結果IDを指定します。

#### Location

DbExAccessは検索結果として複数のレコード・イメージを返します。Locationパラメータは所得する結果の位置を指定します。0オフセットで結果位置を整数で指定します。DbExAccessでは検索結果件数を返しますがその値以上を指定することはできません。

### 戻り値

Btrieveの物理位置が返されます。

### 注意

レコードの物理位置はBtrieveで管理される値で整数のシリアルな値ではありません。

## DbExSetMode

---

Object.DbExSetMode( NewMode As String ) As Integer

### 概要

DbExAccessメソッド/関数にてカレント・レコードを検索対象とするか除外するかを設定します。

### パラメータ

#### NewMode

検索対象にする場合はも列"UC"をパラメータとして設定します。カレント・レコードを検索対象

としない場合は文字列"EG"をパラメータとして設定します。"UC" または "EG"以外の文字列を設定した場合は、DbExAccessメソッド/関数でエラーが発生します。

パラメータ値	動作
UC	カレント・レコードを検索対象に含める。
EG	カレント・レコードを検索対象から除外する。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

## DbFindPercentage

---

```
Object.DbFindPercentage(TableName As String,  
                        KeyName As String,  
                        PhysicalPosition As Long,  
                        Percentage As Integer) As Integer
```

## 概要

レコード位置をパーセントで取得します。

## パラメータ

### TableName

テーブル名を指定します。

### KeyName

インデックス名を指定します。ヌルを指定した場合は物理位置で取得します。

### PhysicalPosition

パーセンテージを得る物理位置を指定します。このパラメータを有効にするためには、KeyNameにヌル文字列を設定します。

### Percentage

取得する位置。たとえば、80パーセントの位置の場合、整数値で8000が返ります。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

03	FILE NOT OPEN
06	Invalid key number
07	Different key number

08	Invalid positioning
09	End of file
22	Data buffer length
4 1	Operation not allowed
4 3	Invalid data record address
82	Lost position

## 注意

Btrieveファイルはバージョン6.X以降の形式でなければ使用できません。

## DbGetByPercentage

---

```
Object.DbGetByPercentage(TableName As String,
                          KeyName As String,
                          Percentage As Integer) As Integer
```

## 概要

パーセント指定でレコードを取得します。

## パラメータ

### TableID

テーブル名を指定します。

### KeyName

インデックス名を指定します。ヌル文字列を指定した場合は物理位置でレコードを取得します。

### Percentage

取得する位置をパーセントで指定します。たとえば、80パーセントの位置の場合、8000を指定します。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

03	FILE NOT OPEN
06	Invalid key number

07	Different key number
08	Invalid positioning
09	End of file
22	Data buffer length
4 1	Operation not allowed
82	Lost position

## 注意

Btrieveファイル形はバージョン6.x以降である必要があります。

## DbGetDataSize

---

```
Object.DbGetDataSize(TableName As String,  
                      FieldName As String) As Integer
```

## 概要

指定したフィールドのデータ・サイズ（バイト）を返します。

## パラメータ

### TableName

テーブル名を指定します。

### FieldName

フィールド名を指定します。

## 戻り値

正常ならばフィールドのデータ・サイズが返されます。負の値はVBManエラー・コード一覧を参照してください。

## DbGetDataType

---

```
Object.DbGetDataType( TableName As String,  
                      FieldName As String ) As Integer
```

## 概要

指定したフィールドのBtrieveデータ型を返します。

## パラメータ

### TableName

テーブル名を指定します。

### FieldName

フィールド名を指定します。

## 戻り値

正常ならばフィールドのデータ型が返されます。負の値はVBManエラー・コード一覧を参照してください。

データ型	リターン値
String	0
Integer	1
Float	2
Date	3
Time	4
Decimal	5
Money	6
Logical	7
Numeric	8
Bfloat	9
Lstring	10
Zstring	11
Note	12
Lvar	13
Unsigned Binary	14
Auto increment	15
Named Index	255

## DbGetDirect

---

```
Object.DbGetDirect(TableName As String,  
                  Pos As Long,  
                  NewIndexName As String) As Integer
```

## 概要

指定された物理レコード位置から、Btrieveデータを読み込みます。

## パラメータ

### TableName

テーブル名を指定します。

### Pos

物理レコード位置を設定します。DbGetPosition関数で取得する、4バイトの整数です。

### NewIndexName

この関数によって得られたレコードの新アクセス・パスをインデックス名で指定します。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

## DbGetFieldData

---

```
Object.DbGetFieldData(TableName As String,  
                      fieldName As String) As String
```

## 概要

データベースのフィールドの値を指定されたテーブルのデータ・バッファから文字列データとして返します。

## パラメータ

### TableName

テーブル名を指定します。

### FieldName

フィールド名を指定します。

## 戻り値

フィールドの値が文字列で返されます。データベースのフィールドから文字列データ型への変換はこの関数内部でおこなわれます。たとえば時間型は、"hh:mm:ss"の形で返されます。エラーが発生した場合は、ヌル文字列が返されます。ヌル文字列が返されるのは、テーブル名、フィールド名がこのメソッドの関連するDDF定義に存在しない場合です。

## DbGetFieldDataByID

---

```
Object.DbGetFieldDataByID(TableID As Integer,
```

## 概要

テーブルID,フィールドIDで指定したデータベースのフィールドの値を文字列データとして返します。

## パラメータ

TableID

テーブルIDを指定します。

FieldID

フィールドIDを指定します。

## 戻り値

フィールドの値が文字列で返されます。データベースのフィールドから文字列データ型への変換はこの関数内部でおこなわれます。たとえば、時間型は、"hh:mm:ss"の形で返されます。エラーが発生した場合は、ヌル文字列が返されます。ヌル文字が返されるのは、テーブルID、フィールドIDがこのメソッドの関連するDDF定義に存在しない場合です。

## 注意

VBMan for Btrieve/VBXバージョン1.xでのテーブル指定、フィールド指定に用いた整数値IDを利用できますので、バージョン1.xのコードを移植する際に便利です。

また、Visual Basic等の32bit言語では文字列型はUNICODEを使っており、当メソッドのパラメータを文字列で指定するDbGetFieldDataメソッドに比べて、内部的に消費するメモリ量が少なく済みます。内部ロジックでデータが保持されている領域を検索する際にも整数値での指定をする当メソッドのほうが比較するバイト数が少なくなりますのでパフォーマンスの向上が期待できます。

## DbGetFieldName

---

Object.DbGetFieldName(TableName As String,  
FieldID As Integer,  
FieldName As String) As Integer

## 概要

テーブル名,フィールドIDで指定したフィールド名を返します。

## パラメータ

TableName

テーブル名を指定します。

FieldID

フィールドIDを指定します。0ベースで指定します。

## FieldName

フィールド名が返されます

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

## DbGetIndexName

---

```
Object.DbGetIndexName(TableName As String,  
                      IndexID As Integer,  
                      IndexName As String) As Integer
```

### 概要

インデックスIDで指定したインデックスが設定されているフィールド名を返します。

### パラメータ

#### TableName

テーブル名を指定します。

#### IndexID

インデックスIDを指定します。0ベースで指定します。

#### IndexName

インデックスが設定されているフィールド名が返されます

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

### 注意

インデックス名 (NamedIndex) はこのメソッドでは得ることができません。

## DbGetModulePath

---

```
Object.DbGetModulePath( BtrieveModulePath As String ) As Integer
```

### 概要

VBManが利用しているBtrieve APIモジュールへのパスを取得します。複数のバージョンのBtrieveが同じパソコンにインストールされている場合、どのモジュールを利用しているかを判別するために使います。

### パラメータ



## BtrieveModulePath

VBManが利用しているモジュールへのパスとモジュール名が返されます。(例: c:\windows\system\wbtrv32.dll)

### 戻り値

正常ならば0が返されます。VBM\_ERR\_GENERIC(-1)が返される場合はBtrieveのモジュールをWin32 APIでロードできない場合です。

### 注意

Btrieve開発キット、Btrieveサーバーに添付されるクライアント・モジュール、VBManに添付されるBtrieve Workstation Engine等複数のBtrieveエンジンがインストールされていてそれぞれバージョンが異なる場合、サーバー上のファイルでもワーク・ステーション・エンジンでオープンされるなどの現象が発生します。そのような場合、Btrieveサーバーのバージョンとクライアント・モジュールのバージョンを合致させる必要があります。当メソッドはそのような状況でのトラブル・シュートのために用意されました。

## DbGetNumOfField

---

```
Object.DbGetNumOfField( TableName As String,  
                        NumOfField As Integer ) As Integer
```

### 概要

指定されたテーブル定義されているフィールド数を返します。

### パラメータ

#### TableName

テーブル名を指定します。

#### NumOfField

定義されているフィールドの数が返されます。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

## DbGetNumOfIndex

---

```
Object.DbGetNumOfIndex ( TableName As String,  
                        NumOfIndex As Integer ) As Integer
```

### 概要

指定されたテーブル定義されているインデックス数を返します。

### パラメータ

#### TableName

テーブル名を指定します。

#### NumOfIndex

定義されているインデックスの数が返されます。セグメント・キーが含まれる場合はその構成メンバーの数も加算された値が返されます。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

---

#### DbGetNumOfRecords

```
Object.DbGetNumOfRecords( TableName As String,  
                          NumOfRec As Long ) As Integer
```

### 概要

指定されたテーブルに存在するレコード数を返します。

### パラメータ

#### TableName

テーブル名を指定します。

#### NumOfRec

レコード数が返されます。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。この関数内部ではBtrieveのstatオペレーションを発行します。

---

#### DbGetNumOfTable

```
Object.DbGetNumOfTable(NumOfTable As Integer) As Integer
```

### 概要

現在読みこんでいるDDFに存在するテーブル数を返します。

### パラメータ

## NumOfTable

テーブル数が返ります。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

## DbGetPosBlock

---

```
Object.DbGetPosBlock( TableName As String,  
                    PosBlock(0 To 127) As Byte) As Integer
```

### 概要

指定されたテーブルに関連するBtrieveのポジション・ブロックを返します。

### パラメータ

#### TableName

テーブル名を指定します。

#### PosBlock

BtrieveのPosBlockを保持するByte型の配列を指定します。配列のサイズはBtrieveの仕様により128バイトを割振る必要があります。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

### 注意

ポジション・ブロックはBtrieveファイルがオープンされている時にのみ有効となりますのでこのメソッドを呼び出す際にはBtrieveファイルがオープンされていることが必須となります。ポジション・ブロックはBtrieveが管理する領域なので通常のアプリケーションではデータはセットできません。このメソッドで得られるポジション・ブロックを利用してBtrieve APIを発行し、DDFに合致しないようなデータを登録した場合は他のコントロールやメソッドでのオペレーションに障害が出る可能性があり、弊社では動作を保証できませんので、Btrieveのデータ型、オペレーション、プログラミングを十分理解した上でのご利用をお願いします。

## DbGetPosition

---

```
Object.DbGetPosition(TableName As String) As Long
```

### 概要

指定されたテーブルの現在のレコードの物理位置を返します。

## パラメータ

TableName

テーブル名を指定します。

## 戻り値

正常ならば物理レコード位置(4バイト) が返されます。テーブルIDの誤り、データベースがオープンされていない場合は-1が返されます。

## 注意

戻り値はシリアルな値ではなく、Btrieveで管理されるユニークな値です。整数値でレコードを識別したい場合は、AutoIncrement型のフィールドを利用します。

## DbGetRecord

---

```
Object.DbGetRecord( TableName As String, _  
                    ArrayOfData() As String ) As Integer
```

## 概要

配列に現在のレコードの内容を取得します。大量の複数のフィールドをDbGetFieldDataメソッドの呼び出し得る方法に比較して実行のパフォーマンス向上が期待できます。

## パラメータ

TableName

テーブル名を指定します。

ArrayOfData

String型の配列を指定します。配列のインデックスはフィールドIDになります。

## 戻り値

正常終了の場合は0が返されます。負の値に関してはVBManエラー・コード一覧をご参照ください。

## 注意

配列のインデックスがDDFで定義されるフィールド数以内で定義してください。フィールドIDはベースが0なのでデータを受け取るString型の配列のベースも0にすると最初のフィールドから順にデータを得ることができます。

## サンプル・コード

```
Private Sub GetFirstRecord()
```

```

Dim rc As Integer, l As Integer
Dim f(0 To 20) As String

rc = Db1.DbAccess(BTR_GET_FIRST, "TEST1", "PRIM_KEY")
If rc <> 0 Then
    MsgBox "エラー " & CStr(rc)
    Exit Sub
End If

rc = Db1.DbGetRecord("TEST1", f)
If rc <> 0 Then
    MsgBox "エラー " & CStr(rc)
    Exit Sub
End If

For l = Lbound(f) To Ubound(f)
    Debug.Print f(l), " ";
Next l
Debug.Print ' 改行

End Sub

```

## DbGetRecordImage

---

```

Object.DbGetRecordImage( TableName As String,
                        ByteArray() As Byte ) As Integer

```

### 概要

指定されたテーブルのレコード・イメージをバイト型の配列に取得します。レコード・イメージとはBtrieveのGET系のオペレーションで取得できるメモリの領域です。

### パラメータ

#### TableName

テーブル名を指定します。

#### ByteArray

レコード・イメージを取得するByte型の配列を指定します。配列のローワー・バウンド値に関係なく、レコード・イメージはゼロ・オフセットから設定されます。配列のサイズを決定する際にレコード長が必要になる場合はDbGetRecordLengthメソッドで取得することが可能です。Delphiをホスト言語としてご利用の場合はOleVariant型でvarByteの配列を指定してください。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

### サンプル・コード

```
Dim rc As Integer
Dim b(0 To 32) As Byte

Rc = VBManDb1.DbGetRecordImage("TEST1",b)
If rc <> 0 Then
    MsgBox "エラー " & CStr(rc)
    Stop
End If
Debug.Print b(0),b(1),b(2)
```

### DbGetRecordLength

---

```
Object.DbGetRecordLength( TableName As String,
                          RecLen As Integer ) As Integer
```

#### 概要

指定されたテーブルに関連するBtrieveファイルのレコード長をバイト単位で返します。

#### パラメータ

TableName

テーブル名を指定します。

RecLen

レコード長が返される2バイト長の整数を指定します。

#### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

#### 注意

当メソッドが呼び出される時点でDDFが読み込まれている必要があります。

### DbGetTableName

---

```
Object.DbGetTableName( TableID As Integer,
                       TableName As String )As Integer
```

#### 概要

テーブルIDを指定してテーブル名を取得します。

## パラメータ

### TableID

テーブルIDを指定します。0ベースの値を指定します。

### TableName

テーブル名が返されます。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

## 注意

当メソッドが呼び出される時点でDDFが読み込まれている必要があります。

### DblsNull

---

```
Object.DblsNull(TableName As String,  
                FileName As String ) As Boolean
```

## 概要

指定されたフィールドのヌル状態を返します。True時にはフィールドはヌルです。

## パラメータ

### TableName

テーブル名を指定します。

### FieldName

フィールド名を指定します。

## 戻り値

フィールドのヌル状態を返します。指定したフィールドが"not null"制約がある場合にはFalseを返します。

### DblsOpen

---

```
Object.DblsOpen( TableName As String ) As Integer
```

## 概要

指定されたテーブルに関連するBtrieveファイルのオープン状態を返します。

## パラメータ

### TableName

テーブル名を指定します。

## 戻り値

オープンしているなら値1が返されます。オープンしていない場合は0を返します。負の値はVbManエラー・コード一覧を参照してください。

## DbLoadDDF

---

Object.DbLoadDDF() As Integer

### 概要

DDFを読み込みます。

### パラメータ

なし

## 戻り値

正常ならば0が返されます。負の値はVbManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

## 注意

## DbLogIn

---

Object.DbLogIn(URI As String) As Integer

### 概要

Action Zen 接続からログアウトします。

### パラメータ

データベースURIを文字列で指定します。詳細はAction Zen8マニュアルをご参照ください。"btrv://localhost/mydb?pwd=hello&dbfile=test1.mkd"のような形式になります。

## 戻り値

正常ならば0が返されます。負の値はVbManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。



## DbLogout

---

Object.DbLogout() As Integer

### 概要

Action Zen 接続からログアウトします。

### パラメータ

なし

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

### 注意

このメソッドを呼び出す前にDbLoginメソッドでログインする必要があります。

## DbOpen

---

Object.DbOpen( TableName As String ) As Integer

### 概要

指定されたテーブルに関連するBtrieveファイルをオープンします。

### パラメータ

TableName

テーブル名を指定します。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

### 注意

このメソッドを呼び出す時はBtrieveファイルはクローズしている必要があります。オープン・モードはこのメソッドが関連しているVBManデータベース・コントロールのOpenModeプロパティによって指定されます。OwnerNameについても同様です。

## DbOpenAll

---

Object.DbOpenAll() As Integer

### 概要

DDFに定義されたBtrieveファイルをすべてオープン状態にします。

### パラメータ

なし。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

### 注意

オープン・モードはこのメソッドが関連しているVBManデータベース・コントロールのOpenModeプロパティによって指定されます。OwnerNameについても同様です。DDFに定義されているファイルにすでにオープン中のものについてはOpenオペレーションは実行されません。また、メソッドはオープン中のエラーを返すこともありません。

---

## DbReset

Object.DbReset() As Integer

### 概要

Btrieveリセット・オペレーションを発行します。

### パラメータ

なし。

### 注意

リセット・オペレーションはオープン中のファイルをすべてクローズするのですでにアプリケーションでオープンしているファイルが存在する場合には注意が必要です。リセット・オペレーションの詳細についてはBtrieveのマニュアルをご参照ください。

---

## DbSetFieldData

Object.DbSetFieldData(TableName As String,  
                        FieldName As String,  
                        Data As String) As Integer

### 概要

Btrieveデータベースへ登録するフィールドのデータを指定されたテーブルのデータ・バッファに設定します。

## パラメータ

### TableName

テーブル名を指定します。

### FieldName

フィールド名を指定します。

### Data

フィールドの値を文字列で指定します。文字列型データからデータベースのフィールドのデータ型への変換はこの関数内部でおこなわれます。Integer型などのバイナリ型も文字列で指定します。Date型は、"YY/MM/DD"または"YYYY/MM/DD"の形で指定します。Time型は"HH:MM:SS"の形で指定します。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

## サンプル

```
Dim rc%  
rc% = DbSetFieldData("給与","欠勤","10")  
If rc% <> 0 then  
    ' エラー処理  
End If
```

### DbSetFieldDataByID

---

```
Object.DbGetFieldDataByID(TableID As Integer, _  
                          FieldID As Integer, _  
                          Data As String) As Integer
```

## 概要

テーブルID、フィールドIDで指定したデータベースのフィールドの値を文字列データとして返します。

## パラメータ

### TableID

テーブルIDを指定します。

### FieldID

フィールドIDを指定します。

## Data

フィールドの値を文字列で指定します。文字列型データからデータベースのフィールドのデータ型への変換はこの関数内部でおこなわれます。Integer型などのバイナリ型も文字列で指定します。Date型は、"YY/MM/DD"または"YYYY/MM/DD"の形で指定します。Time型は"HH:MM:SS"の形で指定します。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

## 注意

VBMan for Btrieve/VBXバージョン1.xでのテーブル指定、フィールド指定に用いた整数値IDを利用できますので、バージョン1.xのコードを移植する際に便利です。

また、Visual Basic等の32bit言語では文字列型はUNICODEを使っており、当メソッドのパラメータを文字列で指定するDbSetFieldDataメソッドに比べて、内部的に消費するメモリ量が少なく済みます。内部ロジックでデータが保持されている領域を検索する際にも整数値での指定をする当メソッドのほうが比較するバイト数が少なくなりますのでパフォーマンスの向上が期待できます。

## サンプル・コード

```
Const TID_EMPLOY = 10
Const FID_EMPLOY_AGE = 2
Dim rc%
rc% = DbSetFieldDataByID(TID_EMPLOY,FID_EMPLOY,"34")
If rc% <> 0 then
    ' エラー処理
End If
```

## DbSetFileName

---

```
Object.DbSetFileName( TableName As String,
                    NewFileName As String ) As Integer
```

## 概要

テーブルに関連するBtrieveファイル名を指定します。DDFビルダーによる定義を実行時に変更します。ファイルがオープンされている状態では変更はできません。

## パラメータ

### TableName

テーブル名を指定します。

### NewFileName

Btrieveファイル名を指定します。ドライブ、パスまで含めることができます。ドライブ、パスを省略した場合はDDFが存在するディレクトリにあるBtrieveファイルを扱います。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

## DbSetLockBias

---

Object.DbSetLockBias( BiasValue As Integer ) As Integer

## 概要

ロック・バイアス値を指定します。当メソッドで指定したロック・バイアス値はこのメソッド呼び出し移行のDbAccess/DbGetDirectメソッドに反映されます。DbAccessメソッドの第一パラメータにロック・バイアス値を毎回加算するコードを記述する必要がなくなります。

## パラメータ

### BiasValue

ロック・バイアス値を指定します。以下の値が指定可能です。ロック動作の詳細はBtrieveのマニュアル記載をご参照ください。

0	通常の状態
100	シングルウェイトロック
200	シングルノーウェイトロック
300	マルチウェイトロック
400	マルチノーウェイトロック

## 戻り値

正常終了の場合は0が返されます。負の値に関してはVBManエラー・コード一覧をご参照ください。

## サンプル・コード

```
Dim rc As Integer
Const TableName = "商品"
Const IndexName = "商品コード"
```

```
With VBMan
```

```
rc = .DbSetLockBias(400) ' multi no wait lock
If rc <> 0 Then Stop
```

‘全レコードをロックする。

```
rc = .DbAccess(BTR_GET_FIRST,"商品","商品コード")
Do
  If rc <> 0 Then Exit Do
  Rc = .DbAccess(BTR_GET_NEXT,"商品","商品コード")
Loop
```

```
rc = .DbSetLockBias(0) ' バイアス解除
Debug.Print rc
rc = .DbUnlock("商品",-2) ' ロック解除
Debug.Print rc
End With
```

## DbSetRecord

---

```
Object.DbSetRecord( TableName As String, _
  ArrayOfData() As String) As Integer
```

### 概要

配列設定したデータを現在レコードのバッファに設定します。大量の複数のフィールドをDbSetFieldDataメソッドの呼び出し得る方法に比較して実行のパフォーマンス向上、コードの簡略化が期待できます。

### パラメータ

#### TableName

テーブル名を指定します。

#### ArrayOfData

String型の配列を指定します。配列のインデックスはフィールドIDになります。

### 戻り値

正常終了の場合は0が返されます。負の値に関してはVBManエラー・コード一覧をご参照ください。

### 注意

配列のインデックスがDDFで定義されるフィールド数以内で定義してください。フィールドIDはベースが0なので先頭フィールドからデータをセットしたい場合は配列のベースも0に設定します。

## サンプル・コード

```
Private Sub InsertRecord()  
Dim rc As Integer, I As Integer  
Dim f(0 To 20) As String  
  
For I = Lbound(f) To Ubound(f)  
    f(I) = Text(I).Text ' Textはコントロール配列  
Next I  
  
rc = Db1.DbSetRecord("TEST1",f)  
If rc <> 0 Then  
    MsgBox "エラー " & CStr(rc)  
    Exit Sub  
End If  
  
rc = Db1.DbAccess(BTR_INSERT, "TEST1", "PRIM_KEY")  
If rc <> 0 Then  
    MsgBox "エラー " & CStr(rc)  
    Exit Sub  
End If  
End Sub
```

## DbSetRecordImage

---

```
Object.DbSetRecordImage(TableName As String, _  
    ByteArray() As Byte) As Integer
```

### 概要

Btrieveのレコード・イメージをByte型の配列で設定します。

### パラメータ

#### TableName

レコード・イメージを設定する対象となるテーブル名を文字列で指定します。

#### ByteArray

設定するレコード・イメージ・データを保持するByte型の配列を指定します。配列は0ベースでなくても配列のローワー・バウンドからゼロ・オフセットでレコード・イメージの内部データ・バッファに設定されます。Delphiの場合はOleVariant型でvarByte型の配列を指定します。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

### 注意

レコード・イメージはDDFで定義されたBtrieveデータ型に従って1バイトアライメントで保持されています。Btrieveデータ型についてはBtrieveのマニュアル等を参照してください。レコード・イメージ・データを保持する配列の上限を設定するにはレコード長が必要になる場合がありますが、レコード長はDbGetRecordLengthメソッドにて取得することが可能です。

### サンプル・コード

```
Dim rc As Integer
Dim length As Integer
Dim b(0 To 30) As Byte

b(0) = 33
b(1) = 34
b(2) = 35

rc = VBManDb1.DbSetRecordImage("TEST1", b)
Debug.Print rc, b(0), b(1), b(2)
```

### DbTransferFromControl

---

```
Object.DbTransferFromControl(TableName As String) As Integer
```

### 概要

現在のフォームにあるデータベース・コントロールのデータをBtrieveとのデータ入出力バッファに転送します。

### パラメータ

TableName  
テーブル名を指定します。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

### 注意

データ転送の対象となるコントロールは、エディット、コンボボックス、リスト、オプション・ボタン、チェック・ボックスです。



## DbTransferToControl

---

Object.DbTransferToControl(TableName As String) As Integer

### 概要

現在のレコードをフォームにあるVBManコントロールの該当フィールドに転送します。

### パラメータ

TableName

テーブルのIDを整数値で指定します。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードが返されます。

### 注意

データ転送の対象となるコントロールは、エディット、コンボボックス、リスト、オプション・ボタン、チェック・ボックスです。

## DbUnlock

---

Object.DbUnlock(TableName As String,  
UnlockType As Integer) As Integer

### 概要

指定されたテーブルのレコード・ロックを解除します。

### パラメータ

TabIdName

テーブル名を指定します。

UnlockType

以下の値が有効です。

アンロックタイプ	詳細
0	シングル・レコード・ロックを解除する
-1	マルチ・レコード・ロックされている現在のレコードのみロックを解除する。
-2	マルチ・レコード・ロックのすべてを解除

### 戻り値

正常ならば0が返されます。テーブルIDの誤り、データベースがオープンされていない場合は-1が返されます。

## バージョン1.Xデータ・アクセス関数

この章は、VBMan for Btrieve/VBXバージョン1.xアプリケーションをVBMan Controls for Btrieveへの移行を容易にするために提供されるバージョン1.xデータ・アクセス関数についてのリファレンスです。

これらの関数の宣言はインストールディレクトリの下 samplesディレクトリにあるファイル VBManV1.BASに用意されています。Btrieveの仕様に依存している部分が多いため、詳細はBtrieveプログラマーズ・マニュアルを参照してください。

関数名	概要
DbAbortTransaction	トランザクションの中止
DbAccess	Btrieve機能の呼び出し
DbAttachControl	バージョン1.xコンパチブル関数が参照するVBManデータベース・コントロールを設定します。バージョン1.xコンパチブル関数の中で一番最初に実行すべき関数です。
DbBeginTransaction	トランザクションを開始
DbClearControlData	データベース・フィールド・コントロールのデータを空にする。
DbClearFieldBuffer	データベースのバッファを初期化
DbEndTransaction	トランザクションを終了
DbFindPercentage	現在のレコード位置をパーセントで得る。
DbGetByPercentage	パーセント指定でレコードを取得する。
DbGetDataSize	データ・サイズを得る
DbGetDataType	データ型を得る
DbGetDirect	物理位置を指定して、レコードを得る。
DbGetFieldData	フィールドのデータをデータベースのバッファから得る
DbGetFieldName	フィールド名を得る。
DbGetPosition	現在レコードの物理位置を得る。
DbGetVersion	現在使用中のBtrieveのバージョンを得る。
DbSetFieldData	フィールドのデータをデータベースのバッファにセット
DbSetFormattedString	リスト・ボックス、コンボ・ボックスにフォーマットされた文字列を設定

DbTransferFromControl	データベース・フィールドの入力値をBtrieveのレコードに転送する。
DbTransferToControl	Btrieveのデータ・バッファの内容をデータベース・フィールド・コントロールに転送する。
DbUnlock	レコード・ロックの解除

以下はBtrieveのExtendedオペレーションに対応したデータベース・アクセス関数です。

関数名	概要
DbExAccess	ExtendedBtrieveオペレーションの実行
DbExGetFieldData	DbExAccess関数による検索結果をフィールド単位で取得する。
DbExGetPosition	DbExAccess 関数による検索結果の物理レコード位置を取得する。
DbExFree	DbExAccess関数の結果を解放する。
DbExSetFormattedString	拡張リスト・ボックス、拡張コンボ・ボックスのFormatイベント内で整形された文字列を設定する。

## DbAbortTransaction

DbAbortTransaction() As Integer

### 概要

トランザクションの中止を宣言します。

### パラメータ

なし。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのエラー・コードが返されます。

## DbAccess

DbAccess(BtrieveOpCode As Integer,  
TableID As Integer,  
KeyID As Integer) As Integer

### 概要

BtrieveOpCodeで指定されるBtrieveの機能呼び出します。

## パラメータ

### BtrieveOpCode

Btrieveのオペレーション・コードを指定します。詳細はBtrieveプログラマーズ・マニュアルを参照。VBManに添付されるBTRIEVE.BASファイルにも定義されています。

### TableID

テーブルのIDを整数値で指定します。データベース・コントロールのDbTableIDプロパティに設定する値を指定します。

### KeyID

Btrieveオペレーションに関連するキーのIDを指定します。IDはVBMan 日本語DDFビルダーで指定した値から1を引いたものを指定します。Btrieveではセグメント・キーがある場合はキーのIDがシリアルではなくなりますが、VBManでは日本語DDFビルダーで決定されるシリアルな値を指定します。

## 戻り値

正常終了ならば0が返ります。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのステータス・コードです。

## サンプル

```
Dim rc%
```

```
Const EMP_TABLE = 10
```

```
Const EMP_ID = 0
```

```
Const EMP_PRIMARY = 0
```

```
rc% = DbSetFieldData(EMP_TABLE,EMP_ID,"066217")
```

```
rc% = DbAccess(BTR_GET_EQUAL,EMP_TABLE,EMP_PRIMARY)
```

```
If rc% <> 0 Then
```

```
    MsgBox "Btrieveの呼び出しに失敗しました" + Str$(rc%)
```

```
    Exit Sub
```

```
End If
```

```
'読み込んだ結果をフォーム上のコントロールに転送
```

```
rc% = DbTransferToControl(EMP_TABLE)
```

```
DbAttachControl
```

---

```
DbAttachControl( ControlName As String,  
                FormName As String ) As Integer
```

## 概要

バージョン1.xコンパチブル関数を使用可能な状態にします。

## パラメータ

### ControlName

この関数呼び出しが成功した以降に呼び出すバージョン1.xコンパチブル関数が参照するDDFディレクトリが設定されているVBManデータベース・コントロールのNameプロパティの値を設定します。

### FormName

上で説明したVBManデータベース・コントロールがにおいてあるフォームの名前を指定します。

## 戻り値

正常終了の場合0を返します。負の値はVBManエラー・コード一覧を参照してください。

## 注意

当関数の呼び出し正常終了後、VBManデータベース・コントロールがにおいてあるフォームをアンロードした場合、それ以降のバージョン1.xコンパチブル関数の呼び出しは動作が保証できません。この関数で指定したVBManデータベース・コントロールがアンロードされた以降にバージョン1.xコンパチブル関数を使う場合は、再度この関数を呼び出して別の現在参照可能なVBManデータベース・コントロールを指定してください。

## DbBeginTransaction

---

DbBeginTransaction(LockBias As Integer) As Integer

## 概要

トランザクションの開始を宣言します。

## パラメータ

### LockBias

トランザクション・ロックを指定。値は100,200,300,400を指定可能。詳細はBtrieveプログラマーズ・マニュアルを参照してください。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのエラー・コードが返されます。

## 注意

トランザクションを開始したらBtrieveファイルをクローズする前にトランザクションの終了または中断を実行してください。

## DbClearControlData

---

DbClearControlData(TableID As Integer) As Integer

### 概要

アクティブなフォームにあるコントロールのデータ<sup>18</sup>を初期化します。Btrieveデータ入出力に使われるバッファは初期化されません。

### パラメータ

TableID

データを初期化したいコントロールが関連しているテーブルのID

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

### コンパチビリティ

バージョン1.xからはパラメータとしてTableIDの指定が追加されました。

## DbClearFieldBuffer

---

DbClearFieldBuffer( TableID As Integer ) As Integer

### 概要

指定したテーブルのデータ・バッファを初期化します。

### パラメータ

TableID

テーブルのIDを整数値で指定します。データベース・コントロールのDbTableプロパティに設定する値を指定します。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

## DbEndTransaction

---

DbEndTransaction() As Integer

### 概要

トランザクションの終了を宣言します。

---

<sup>18</sup>

VBManエディットのテキスト・プロパティを空にします。

## パラメータ

なし。

## 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのエラー・コードが返されます。



DbExAccess( OpCode As Integer,  
TableID As Integer,  
SearchCondition As String,  
Fields As String,  
IndexID As Integer,  
SkipRecords As Integer,  
MaxRecords As Integer,  
IgnoreCase As Integer,  
ResultID As Integer,  
NumberOfResults As Integer ) As Integer

## 概要

BtrieveのExtended関数を実行します。

## パラメータ

### TableID

テーブルのIDを整数値で指定します。

### OpCode

Extendedオペレーション・コードを指定します。指定できる値はBTRIEVE.BASまたはVBManV1.BASに宣言してある以下の値です。

BTR_GET_PREVIOUS_EXTENDED	37
BTR_GET_NEXT_EXTENDED	36
BTR_STEP_PREVIOUS_EXTENDED	39
BTR_STEP_NEXT_EXTENDED	38

### SearchCond

検索条件文字列を指定します。拡張リスト・ボックス、拡張コンボ・ボックスの検索条件文字列と同じ形式で指定します。

フィールドは@をプリフィックスと比較式の左辺に指定します。（例；フィールドIDの10は@10と指定する）比較演算子は以下を指定します。

=	等しい
>	より大きい
<	より小さい

<>	等しくない
>=	より大きいか等しい
<=	より小さいか等しい

複合検索をする場合は検索条件を& (AND)または| (OR)で結合します。比較対象を即値で指定する場合はスペースがディリミッタになります。スペースが含まれる即値を指定する場合はシングル・クォート、ダブル・クォート、スラッシュで文字列を囲みます。以下に検索文字列の例を示します。

```
@0 >= 100 & @1 <= 1000
@1 = "VBMan" | @1 = "Visual Basic"
@2 <> /Microsoft/ & @2 <> /Borland/
```

### Fields

検索結果として抽出するフィールドを指定します。フィールドはIDをカンマで区切った文字列を指定します。フィールドIDはDDFビルダーの表示値を0オフセットにした値です。(例: 10,22,34)

### IndexID

GET NEXT/PREVIOUS EXTENDEDオペレーションを実行する際のインデックスのパスを指定します。DbAccess関数の第三パラメータと同じ形式で指定します。カレント・レコードが確立されていない場合、GET NEXT/PREVIOUS EXTENDEDオペレーションはエラーを返します。

### SkipRecords

検索条件を満たさない場合にスキップするレコード数を指定します。

### MaxRecords

検索結果の最大レコード件数を指定します。Btrieveでは結果が65,536バイトをこえることができません。65,535をFieldsで指定した抽出するフィールドのデータ長の合計値で割った数以上は設定できません。

### IgnoreCase

文字列フィールドを比較する場合、0以外の値を指定すると英数字に関して大文字、小文字の区別無く比較を実行します。2バイト文字にはこのオプションはサポートされません。

### ResultID

結果IDが整数値で返されます。結果を取得するにはこのIDを指定してDbExGetFieldData,DbExGetPosition関数を使用します。結果が不要になったらこのIDでDbExFree関数を呼び出し資源を解放します。

### NumberOfResults

検索結果件数が返されます。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。これらのプログラミング・レベルのエラーは関数内部からメッセージ・ボックスで詳細が表示されます。整数値が返された場合はBtrieveからのエラー・コードです。また、エラー9,22,60,64,65等が返った場合でも結果が存在する場合もあることに注意してください。

以下にこのオペレーションでBtrieveから返されるエラー・コードを抜粋します。

03	FILE NOT OPEN
06	Invalid Key Number
07	Different Key Number
08	Invalid Position
09	End Of File
22	Data Buffer Length
60	Reject Count Reached
61	Work Space Too Small
62	Incorrect Descriptor
6	Filter Limit Reached
4	
65	Incorrect Field Offset
82	Lost Position

### 注意

Btrieveファイル形式は6.x/7.xを使用してください。

### サンプル

```
Dim rc As Integer
Dim ResultID As Integer
Dim NumOfRecords As Integer
Dim Fields As String
Dim SearchCond As String
Dim a$,l As Integer
```

```
SearchCond = "@1 > 100 & @1 < 200"
```

```
Fields = "0,1,2,3"
```

```
rc = DbExAccess( BTR_GET_NEXT_EXTENDED, _
                0, _
```

```

        SearchCond, _
        Fields, _
        0, _
        127, _
        100, _
        0, _
        ResultID, _
        NumOfRecords )
If rc <> 9 & rc <> 0 & rc <> 60 Then
    MsgBox "Btrieve Extended Opeartion Error " & Str$(rc)
    Stop
End If
' 結果をリスト・ボックスに表示
For I = 0 To NumOfRecords -1
    a$ = DbExGetFieldData(ResultID,0,I)
    a$ = a$ & " " & DbExGetFieldData(ResultID,1,I)
    ResultListBox.AddItem a$
Next I
' 結果を保持するメモリを解放
rc = DbExFree(ResultID)

```

DbExFree( ResultID As Integer ) As Integer

### 概要

DbExAccess関数によって返される結果に付随する資源を解放します。

### パラメータ

ResultID

DbExAccess関数の結果として返される結果IDを指定します。

### 戻り値

ResultIDが不正の場合に-1が返されます。正常に資源を解放できた場合は0を返します。

### 注意

DbExAccess関数の結果は検索結果件数と抽出するフィールドのサイズと件数に比例してメモリを消費します。不要になった結果はこの関数を呼び出してかならず解放してください。

## DbExGetFieldData

---

DbExGetFieldData( ResultID As Integer,  
FieldID As Integer,  
Location As Integer) As String

### 概要

DbExAccess関数の結果をフィールド単位で取得します。

### パラメータ

ResultID

DbExAccess関数の結果として返される結果IDを指定します。

FieldID

フィールドの値を整数値で指定します。データベース・コントロールのDbFieldIDプロパティに設定する値を指定します。VBMan DDFビルダーで表示されるのフィールドIDを0オフセットした値です。

Location

DbExAccessは検索結果として複数のレコード・イメージを返します。Locationパラメータは所得する結果の位置を指定します。0オフセットで結果位置を整数で指定します。DbExAccessでは検索結果件数を返しますがその値以上を指定することはできません。

### 戻り値

パラメータ不正、メモリが不足してVisual Basic文字列が作成できない場合にヌル文字列を返し

ます。正常の場合はフィールドの値を文字列で返します。

## DbExGetPosition

---

DbExGetPosition( ResultID As Integer,  
Location As Integer ) As Long

### 概要

DbExAccess関数の結果を物理レコード位置単位で所得します。

### パラメータ

#### ResultID

DbExAccess関数の結果として返される結果IDを指定します。

#### Location

DbExAccessは検索結果として複数のレコード・イメージを返します。Locationパラメータは所得する結果の位置を指定します。0オフセットで結果位置を整数で指定します。DbExAccessでは検索結果件数を返しますがその値以上を指定することはできません。

### 戻り値

Btrieveの物理位置が返されます。

### 注意

レコードの物理位置はBtrieveで管理される値で整数のシリアルな値ではありません。

## DbExSetFormattedString

---

DbExSetFormattedString( FldData As String ) As Integer

### 概要

拡張リスト・ボックス・コントロール、拡張コンボ・ボックス・コントロールのFormatイベント内で整形された文字列を設定します。

### パラメータ

#### FldData

拡張リスト・ボックス・コントロール、拡張コンボ・ボックス・コントロールに挿入する文字列を指定します。最大長2,048バイトまでの文字列を指定できます。

### 戻り値

正常ならば0が返ります。設定する文字列の長さが2,048バイト以上の場合は-1が返されます。

### サンプル

以下はリスト・ボックスのFormatイベントのサンプルです。FormatイベントにはDbTabStopsプロパティを設定した場合、複数の項目データがタブ文字で区切られて渡されてきます。以下の例ではその先頭の項目だけをリスト・ボックスに表示する例です。

```
Sub VBManExListBox1_Format( FldData As String )
Dim rc As Integer, Formatted$ As String
Dim Pos As Integer

Pos = Instr( FldData, chr$(9) )
If Pos > 0 Then
    Formatted$ = Left$(FldData,Pos - 1 )
    rc = DbSetFormattedString( Formatted$ )
End If
End Sub
```

DbFindPercentage(TableID As Integer,  
KeyNo As Integer,  
PhysicalPosition As Long,  
Percentage As Integer) As Integer

### 概要

レコード位置をパーセント指定で取得します。

### パラメータ

#### TableID

テーブルのIDを整数値で指定します。

#### KeyNo

キー番号を指定します。-1を指定した場合は物理位置で取得します。

#### PhysicalPosition

パーセンテージを得る物理位置を指定します。このパラメータを有効にするためには、KeyNoを-1に設定します。

#### Percentage

取得する位置。たとえば、80パーセントの位置の場合、整数値で8000が返ります。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。エラーの場合はBtrieveのステータス・コードがそのまま返されます。

03	FILE NOT OPEN
06	Invalid key number
07	Different key number
08	Invalid positioning
09	End of file
22	Data buffer length
4 1	Operation not allowed
4 3	Invalid data record address
82	Lost position

### 注意

Btrieveファイルはバージョン6.X以降の形式でなければ使用できません。



## DbGetByPercentage

---

DbGetByPercentage( TableID As Integer,  
KeyNo As Integer,  
Percentage As Integer) As Integer

### 概要

パーセント指定でレコードを取得します。

### パラメータ

#### TableID

テーブルのIDを整数値で指定します。

#### KeyNo

キー番号を指定します。-1を指定した場合は物理位置でレコードを取得します。

#### Percentage

取得する位置をパーセントで指定します。たとえば、80パーセントの位置の場合、8000を指定します。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外の値はBtrieveのステータス・コードがそのまま返されます。

03	FILE NOT OPEN
06	Invalid key number
07	Different key number
08	Invalid positioning
09	End of file
22	Data buffer length
41	Operation not allowed
82	Lost position

### 注意

Btrieveファイルの形式はバージョン6.x以降であることが必要です。詳細はBtrieve Programmer's Manualを参照してください。

## DbGetDataSize

---

DbGetDataSize(TableID As Integer,  
FieldID As Integer) As Integer

### 概要

指定したフィールドのデータ・サイズ（バイト）を返します。

### パラメータ

TableID

テーブルのIDを整数値で指定します。

FieldID

フィールドのIDを整数値で指定します。

### 戻り値

正常ならば指定されたフィールドのデータ・サイズが返されます。負の値はVBManエラー・コード一覧を参照してください。

### 注意

なし。

DbGetDataType

---

DbGetDataType(TableID As Integer,  
FieldID As Integer) As Integer

### 概要

指定したフィールドのデータ型を返します。

### パラメータ

TableID

テーブルのIDを整数値で指定します。

FieldID

フィールドのIDを整数値で指定します。

### 戻り値

正常ならばDDFで定義されるデータ型のIDが返されます。負の値はVBManエラー・コード一覧を参照してください。

データ型	データ型ID
String	0
Integer	1

Float	2
Date	3
Time	4
Decimal	5
Money	6
Logical	7
Numeric	8
BFloat	9
LString	10
ZString	11
Note	12
LVar	13
U n s i g n e d Binary	14
Auto increment	15
Named Index	255

**注意**  
なし。

DbGetDirect(TableID As Integer,  
Pos As Long,  
NewIndex As Integer) As Integer

### 概要

指定された物理レコード位置から、レコードを読み込みます。コントロールに入力されたデータをBtrieveのレコードに転送します。

### パラメータ

#### TableID

テーブルのIDを整数値で指定します。

#### Pos

物理レコード位置を設定します。DbGetPosition関数で取得する、4バイトの整数です。

#### NewIndex

この関数によって得られたレコードの新アクセス・パスをインデックス番号で指定します。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外はBtrieveのエラー・コードが返されます。

DbGetFieldData(TableID As Integer,  
FieldID As Integer) As String

### 概要

データベースのフィールドの値を指定されたテーブルのデータ・バッファから文字列データとして取得します。すべてのBtrieveデータ型は文字列型として返されます。バイナリ・データとして取得する場合はDbGetImageメソッドを使います。

### パラメータ

#### TableID

テーブルのIDを整数値で指定します。データベース・コントロールのDbTableIDプロパティに設定する値を指定します。

#### FieldID

フィールドのIDを整数値で指定します。データベース・コントロールのDbFieldIDプロパティに設定する値を指定します。

### 戻り値

フィールドの値が文字列で返されます。データベースのフィールドから文字列データ型への変換はこの関数内部でおこなわれます。したがって、時間型などは、"hh:mm:ss"の形で返されます。エラーが発生した場合は、ヌル文字列が返されます。

テーブルID、フィールドIDの範囲が正しくない場合はこの関数内部からメッセージ・ボックスで通知されます。

### サンプル

```
Dim rc%,EmpNo$
```

```
Const EMP_TABLE = 10
```

```
Const EMP_ID = 0
```

```
EmpNo$ = DbGetFieldData(EMP_TABLE,EMP_ID)
```

```
DbGetFieldName(TableID As Integer,  
              FieldID As Integer) As String
```

### 概要

指定したフィールドIDに対応するDDF定義のフィールド名を返します。

### パラメータ

TableID

テーブルのIDを整数値で指定します。

FieldID

フィールドのIDを整数値で指定します。

### 戻り値

正常ならばフィールド名が返されます。テーブルID,フィールドIDが不正の場合はヌル文字列が返ります。

DbGetPosition(TableID As Integer) As Long

### 概要

指定されたテーブルの現在のレコードの物理位置を返します。

### パラメータ

TableID

テーブルのIDを整数値で指定します。

### 戻り値

正常ならば物理レコード位置が返されます。テーブルIDの誤り、データベースがオープンされていない場合は-1が返されます。

### 注意

戻り値はシリアルな値ではなく、Btrieveが管理するユニークな4バイトの整数値です。

DbGetVersion(Version As BtrieveVersion) As Integer

### 概要

現在使用中のBtrieveのバージョン情報を取得します。

### パラメータ

#### Version

Btrieveバージョンの構造体。VBMANV1.BASに宣言があります。

Type BtrieveVersion

  MajorVersion As Integer

  MinorVersion As Integer

  BtrieveType As String \* 1

End Type

たとえば、Btrieve for Windows バージョン5.10aのWBTRCALL.DLLではMajorVersion=5, MinorVersion=1, BtrieveType="W" となります。詳細はBtrieveプログラマーズ・マニュアルのGET VERSIONオペレーションを参照してください。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。それ以外の値はBtrieveのステータス・コードがそのまま返されます。

### 注意

返される値はVBManのバージョンではないことに注意してください。



```
DbSetFieldData(TableID As Integer,  
                FieldID As Integer,  
                Data As String) As Integer
```

### 概要

データベースへ登録するフィールドのデータを指定されたテーブルのデータ・バッファに設定します。

### パラメータ

#### TableID

テーブルのIDを整数値で指定します。データベース・コントロールのDbTableIDプロパティに設定する値を指定します。

#### FieldID

フィールドのIDを整数値で指定します。データベース・コントロールのDbFieldIDプロパティに設定する値を指定します。

#### Data

フィールドの値を文字列で指定します。文字列からデータベースのフィールドのデータ型への変換はこの関数内部でおこなわれます。したがって、日付け型などは、"YYYY/MM/DD"などの形で指定します。

### 戻り値

正常ならば0が返ります。負の値はVBManエラー・コード一覧を参照してください。

### サンプル

```
Dim rc%  
Const EMP_TABLE = 10  
Const EMP_ID = 0  
rc% = DbSetFieldData(EMP_TABLE,EMP_ID,"100")  
If rc% <> 0 then Stop      ' error process
```

### DbSetFormattedString(Formatted As String) As Integer

#### 概要

リスト・ボックス、コンボ・ボックスのフォーマット・イベント・プロシージャで使用します。フォーマット・イベント発生時にはリスト・ボックスに挿入されるレコード・セットがDbGetFieldData関数を通じて使用可能となります。プロパティで指定したカラムのデータだけでは不十分な場合、それらの形式を整えた文字列をこの関数で指定します。

#### パラメータ

##### Formatted

フォーマットした文字列。

#### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

#### 注意

フォーマット・イベント・プロシージャの中でのみ使用してください。それ以外の使用は結果を保証できません。ヌル文字列が指定された場合にはリスト・ボックスまたはコンボ・ボックスへの挿入をおこないません。

#### サンプル

```
Sub VBManListBox1_Format( FieldData As String )
Dim fmt$,rc As Integer
If Val(FieldData) < 100 Then
    fmt$ = FieldData & " " & DbGetFieldData(0,1)
    rc = DbSetFormattedString(fmt$)
Else
    VBManListBox1.Abort = True
End If
End Sub
```

## DbTransferFromControl

---

DbTransferFromControl(TableID As Integer) As Integer

### 概要

現在のフォームにあるデータベース・コントロールのデータをBtrieveの入出力バッファに転送します。

### パラメータ

TableID

テーブルのIDを整数値で指定します。

### 戻り値

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

### 注意

データ転送の対象となるコントロールは、エディット、コンボボックス、リスト、オプション・ボタン、チェック・ボックスです。

## DbTransferToControl

---

DbTransferToControl(TableID As Integer) As Integer

### 概要

現在のレコードをフォームにあるデータベース・コントロールのデータとして転送します。

### パラメータ

TableID

テーブルのIDを整数値で指定します。

### 戻り値

正常ならば0が返されます。テーブルIDの誤り、データベースがオープンされていない場合は-1が返されます。フォームがアクティブでない場合は-2を返します。

### 注意

データ転送の対象となるコントロールは、エディット、コンボボックス、リスト、オプション・ボタン、チェック・ボックスです。

---

DbUnlock(TableID As Integer,UnlockType As Integer) As Integer

**概要**

指定されたテーブルのレコード・ロックを解除します。

**パラメータ**

TableID

テーブルのIDを整数値で指定します。

UnlockType

以下の値を指定してください。

UNLOCKTYPE	詳細
0	シングル・レコード・ロックを解除する
-1	マルチ・レコード・ロックされているBtrieveファイルの現在のレコードのみロックを解除する。
-2	マルチ・レコード・ロックのすべてを解除

**戻り値**

正常ならば0が返されます。負の値はVBManエラー・コード一覧を参照してください。

**注意**

なし。

## 作成したアプリケーションの配布

---

VBMan Controls for Btrieveでは、DDFディレクトリがカスタム・コントロールのDDFDirプロパティに設定されます。このプロパティには通常ドライブとパス名が記述されることが多く、作成したアプリケーションを配布する場合は動作環境においてDDFが置かれるパスをDDFDirプロパティで指定される値以外に変更する必要が生じると思われます。この章では、DDFDirプロパティを実行時に、VBMAN.INIファイルから指定する方法を解説します。

### DistNameプロパティとVBMan.INIファイル

VBManデータベース・コントロールにはDistNameプロパティがあります。このプロパティにはVBMan.INIファイルの中にDDFへのパスが記述してあるセクション名を指定できます。アプリケーションの名前やEXEファイルの名前などシステムでユニークに識別できる値を設定します。VBMAN.INIファイルのDistNameプロパティで指定されるセクションの中にはVBManデータベース・コントロールのNameプロパティの値に置換するDDFディレクトリを記述します。以下はVBMAN.INIファイルのサンプルです。

```
[StockControlApp]
ItemDb = c:\StockSys\DDF
Transaction = c:\StockSys\DDF
```

```
[SalesReport]
SalesTrn = z:\Sales\DDF
SalesPerson = z:\Sales\DDF
SalesTmp = z:\Sales\Tmp
```

上の例では、StockControlAppには2つのVBManデータベース・コントロールがありNameプロパティはそれぞれItemDb,Transactionが設定されています。配布時にはc:\StockSys\DDFというディレクトリをDDFディレクトリとして参照します。

[SalesReport]には3つのVBManデータベース・コントロールがあり、NameプロパティはそれぞれSalesTrn,SalesPerson,SalesTmpです。参照するDDFディレクトリは最初の2つがz:\Sales\DDFでSalesTmpという名前のコントロールはz:\Sales\TmpディレクトリからDDFを読み込みます。

以上からわかるように、複数の環境でDDFの存在するドライブ、パスが変更される可能性のあるアプリケーションを作成する場合は、DistNameプロパティを設定することが必須で参照するDDFのディレクトリが異なる場合はVBManデータベース・コントロールのNameプロパティの値をシステムでユニークにすることが必要です。この場合のシステムとはVisual Basicでは1つのプロジェクトになります。

---

<sup>19</sup> EXEファイル単位。

## プロファイルへの書き込み

VBMan.INIファイルへの書き込み方法としては、単にテキスト・エディターで編集する方法もありますが、インストーラーを作成する場合はWritePrivateProfileStringというWindows APIを使うのが一般的です。Visual Basic 4.0ではSaveSettingsステートメントが用意されていますが、32bit環境ではレジストリに登録されるためWindows APIを直接呼び出すことが必要です。以下はVisual Basic 4.0 32bitでのプロファイルへの書き込みの例です。

‘モジュール・ファイルで宣言してください。

```
Declare Function WritePrivateProfileString Lib "kernel32" _
    Alias "WritePrivateProfileStringA" _
    (ByVal lpApplicationName As String, _
    ByVal lpKeyName As Any, _
    ByVal lpString As Any, _
    ByVal lpFileName As String) As Long
```

‘以下は書き込みサンプル

```
Dim rc As Long
```

```
Dim DDFDir As String
```

```
DDFDir = "C:\StockSys\DDF"
```

```
rc = WritePrivateProfileString( "StockControlApp", _
    "ItemDB", _
    DDFDir, _
    "vbman.ini" )
```

この結果VBMAN.INIファイルは以下になります。

```
[StockControlApp]
```

```
ItemDB = c:\StockSys\DDF
```

## 注意

VBManデータベース・コントロールはDistNameプロパティが設定されかつ、実行時であれば、DDFDirプロパティに入れ替えのためにVBMan.INIファイルを参照します。VBMan.INIファイルにVBManデータベース・コントロールのNameプロパティと同じエントリがあれば、必ずDDFDirプロパティの値は置換されてしまいます。デバッグ中などこのDistNameプロパティを指定している場合はどちらのDDFが参照されているか、注意してデバッグしてください。

## DDFビルダー・リファレンス

---

Btrieve 6.15 当時からDDF編集環境がないお客様用にVBMan DDFビルダーが添付されています。当時のDDFがAction Zen DDF Builder で読めない場合にご利用ください。VBMan DDFビルダーはBtrieve 6.1X当時に定義されていたBtrieveデータ型のみサポートとなります。

### DDFビルダーの起動

---

VBManメニューからDDFビルダーを選択します。以下のような初期画面が表示されます。メニューは初期状態になります。



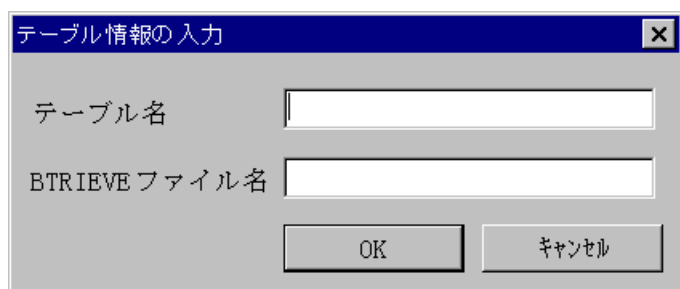
### 新規DDFの作成

---

DDFビルダー起動後の「ファイル」メニューの下の「新規DDFの作成」メニューを選択します。以下のウィンドウが表示されますから、新規にDDFを作成するディレクトリを指定し「開く」ボタン



を押します。以下のテーブル情報入力ダイアログが表示されます。テーブル名とBtrieveファイル名を指定してOKボタンを押します。Btrieveファイル名にはドライブ名やサーバー名にパスを含めて記述することも可能です。単にファイル名を設定した場合にはDDFと同じディレクトリにBtrieveファイルが存在することになります。



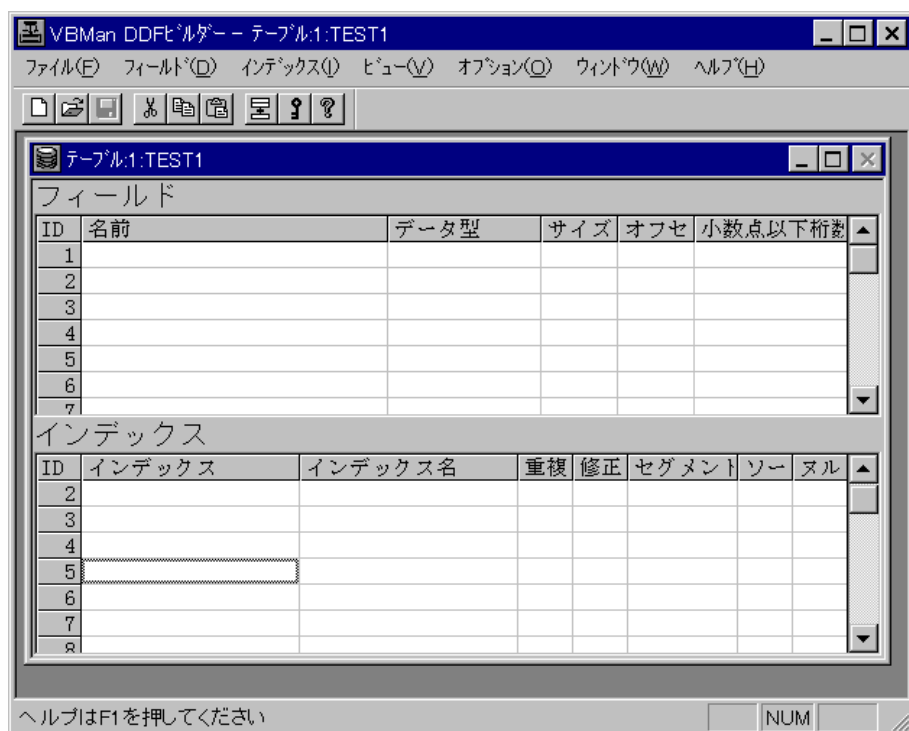
テーブル情報の入力

テーブル名

BTRIEVE ファイル名

OK キャンセル

テーブル情報を入力するとメニューが切り替わり、テーブルを定義できるDDF編集モードのメニューになります。以下は初期画面の様子です。



VBMan DDFエディター - テーブル1:TEST1

ファイル(F) フィールド(F) インデックス(I) ビュー(V) オプション(O) ウィンドウ(W) ヘルプ(H)

テーブル1:TEST1

フィールド

ID	名前	データ型	サイズ	オフセ	小数点以下桁数
1					
2					
3					
4					
5					
6					
7					

インデックス

ID	インデックス	インデックス名	重複	修正	セグメント	ソー	ノル
2							
3							
4							
5							
6							
7							
8							

ヘルプはF1を押してください NUM

## フィールドの定義

DDF編集モードのメニューで「フィールド」メニューから「追加」を選択するとフィールド追加のダイアログが以下のように表示されます。



フィールド設定

フィールド名

データ型

サイズ

小数点以下桁数

登録      登録・継続      キャンセル

フィールド名を入力してデータ型、データ・サイズを入力します。上の例ではString型のフィールドを定義しているので、小数点以下桁数の入力は無効になっています。入力項目が正しいことを確認したら「登録ボタン」を押します。

フィールド設定の入力がフィールド・グリッドに表示されます。以下は例です。

VBMan DDFビルダー - テーブル1:TEST1

ファイル(F) フィールド(F) インデックス(I) ビュー(V) オプション(O) ウィンドウ(W) ヘルプ(H)

テーブル1:TEST1

フィールド

ID	名前	データ型	サイズ	オフセ	小数点以下桁数
1	ID	String	10	0	
2					
3					
4					
5					
6					
7					

インデックス

ID	インデックス	インデックス名	重複	修正	セグメント	ソー	ヌル
2							
3							
4							
5							
6							
7							
8							

ヘルプはF1を押してください

NUM

## インデックスの定義

DDF編集モードのメニュー「インデックス」から追加を選択します。以下のような画面でインデックスの定義をします。

インデックス設定

フィールド ID

ソート・オーダー

昇順  降順

キー・タイプ

重複  ヌル

修正可能  セグメント

セグメント・メンバー

OK キャンセル

フィールドを選択して、ソート・オーダー、キータイプを選択してOKボタンを押します。「データ変換モード」でセグメントがチェックされている場合にはセグメント・メンバーのリストにメンバーとして表示されるだけで、このダイアログは閉じません。セグメントのチェックが無いフィールドが定義されてOKボタンを押すと、すべてのセグメント・メンバーが定義された状態になり、このダイアログはクローズします。

キーの定義はテーブル・ウィンドウのインデックス・グリッドに表示されます。

セグメント・キーを定義する場合にはキー・タイプ、ソート・オーダーがすべてのセグメント・キーのメンバーについて同じでないとBtrieveファイルの生成時にエラーが発生しますのでご注意ください。

セグメント・キーはセグメントのフラグがオフになった項目までが1セグメントとなります。上の例ではIDとITEMNAMEで1つのセグメント・キーが定義されています。

## 既存DDFの編集

---



既存DDFを編集する場合は、DDFビルダー起動画面の「ファイル」メニューから「既存DDFのオープン」を選択します。ファイルを選択するダイアログが表示されますので、FILE.DDF、FIELD.DDF、INDEX.DDFの存在するディレクトリを指定します。DDFビルダーの設定を「データ移行モード」とした場合にはDDFで指定されるディレクトリにデータ移行するBtrieveデータ・ファイルが必要です。この時点でBtrieveデータとDDFの定義が合致しない場合にはDDFビルダーは正常にデータ変換できなくなるのでご注意ください。

## データ変換モードの動作について

システム・オプション・メニューから「Btrieveデータ変換モード」を選択すると、既存のBtrieveファイルに対して追加、削除されたフィールド、インデックスについてはDDFの形式の変更と共に実際のBtrieveデータにもフィールドやが追加・削除され、DDFとの整合性が保たれるように動作します。データ変換モードで動作している場合にはフィールドの修正、およびインデックスの修正メニューはディセーブル状態となり利用できない仕様です。既存のデータ・ファイルのフィールドやインデックスを修正したい場合は、一旦削除してからフィールドやインデックスを追加することになります。

また、重複を許さないインデックスを追加する際には既存のデータに重複が存在するとエラーになってインデックスが生成できなくなりますのでご注意ください。

## テーブルの搬入・搬出

DDF編集モードでは「ファイル」メニューから「テーブルの搬入」、「テーブルの搬出」が実行で

きます。テーブル定義を別のDDFにコピーする場合や一部定義が異なるが大半の項目が同じテーブルを定義する場合に便利です。搬出したテーブルを同じDDFに搬入する場合はDDFに存在しないテーブル名を設定する必要があります。搬出されたファイルのテーブル名を変更する場合は搬出したファイルをnotepad等のエディターで[TableName]の次の行を編集します。(2行目) Btrieveファイル名は[TableLocation]の次の行にあります。(4行目) 編集したファイルは再度、搬入して編集します。

## dbMagicデータ辞書を搬入

---

VBMan DDFビルダーのオプションメニューから「dbMagicテーブルの搬入」を選択することで、dbMagic バージョン 4.26,4.30,7.00のデータ辞書情報をDDFに搬入することが可能です。dbMagicのデータ辞書をDDFに搬入する手順は以下のとおりです。

- ① dbMagicでデータ辞書出力を実行します。このとき、dbMagicの出力オプションの「モード」で、「I」を選択します。このモード以外で出力された場合の動作は保証されませんので、ご注意ください。
- ② VBMan DDFビルダーを起動して、DDF編集の状態にします。新規の場合は最初に適当なテーブル名、Btrieveファイル名を入力して空のテーブルが1つだけの状態に、オプション・メニューから「dbMagicテーブル搬入」を選択し、搬入が完了したら、最初のテーブルをファイル・メニューの下からテーブルの削除を選択し、テーブルを削除します。既存のDDFをオープンした場合にはこのDDFにdbMagicデータ辞書の定義を追加することになります。
- ③ メニューから「dbMagicファイル搬入」を選択すると、ファイル指定ダイアログが表示されます。先程1で出力したデータ辞書の定義ファイルを指定します。
- ④ 搬入が完了すると、dbMagicのBtrieveファイル毎にDDFの編集ウィンドウが表示されます。複数定義を搬入する場合、途中でエラーとなる定義が存在した場合、それ以降の定義の搬入はされませんので、注意してください。

## 注意事項

- ① dbMagicのデータ辞書でメモ型(M)のフィールドはDDFのString型に変換します。VBManでは可変長のレコードはDDFの最後に1フィールドのみサポートとなりますから、複数のメモ型の存在するdbMagicのデータ辞書定義、メモ型のフィールドが最後がないもの等は扱うことができません。
- ② dbMagicのデータ辞書で日付け型(D)のフィールドはDDFのInteger型4バイトに変換します。これは、dbMagicがBtrieveのDate型を使っていないことが原因です。dbMagicでは西暦1

年1月1日を1として起算した数値を日付け型として使っていますので、お客様のアプリケーションでデータを変換するなどの作業が必要になります。

- ③ dbMagicのデータ辞書定義でBtrieveファイル名が指定されないものはVBMan DDFビルダーでは、取り込めません。（出力ファイルのFILEがある行にNAME=の記述がないもの）データ辞書定義でdbMagicの論理パスを指定して作成されたBtrieveファイルはVBMan DDFビルダーのオプション・メニューから「Btrieveファイル情報」で論理パス情報を修正することが可能です。

## Visual Basic用フォームの生成

---

DDF編集モードのメニューで「オプション」からフォーム生成を選択することでVisual Basic用のフォームを生成することが可能です。Visual Basicで簡単なマスター登録ができるフォームを生成することが可能です。画面の解像度は無視されるのでフィールドが多い場合はアプリケーションが動作するパソコンの解像度にあわせてフォームを編集してください。生成されたフォームはVisual Basicでプロジェクトに追加して利用します。

## フィールドIDリストの生成

---

Visual Basicでアプリケーションを作成する場合、VBManのメソッドのパラメータをConstで指定したい場合はこの機能を利用してフィールドIDのリストを生成してそれをVisual Basicのプロジェクトに追加します。

メソッドのパラメータとしてテーブル名やフィールド名を文字列で指定する場合にはフィールドが追加になってもコードの変更は必要無いです。VBManバージョン1.x関数やID指定のメソッド(DbGetFieldDataByID等)を使っている場合、IDを直接指定しているとフィールドの挿入によってIDが変わるとコードの変更も必要になる場合があります。関数やメソッドのパラメータをシンボルで指定しておけば、フィールドの挿入でIDが変わっても、Const定義部分だけを変更するだけで関数やメソッドの呼び出しコードを変更する必要はなくなります。

以下はIDリストの例です。

```
テーブルID for TEST1.DAT
Global Const TID_TEST1 = 0
フィールドID
Global Const FID_ID = 0
Global Const FID_ITEMNAME = 1
Global Const FID_PRICE = 2
インデックスID
```

```
Global Const KID_ID = 0
Global Const KID_ITEMNAME = 1
```

以下は上のIDリストでのVisual Basicコード例です。

```
Dim rc As Integer
With VBManDb1
    Call .DbSetFieldDataByID(TID_TEST1,FID_ID,"101")
    Call .DbSetFieldDataByID(TID_TEST1,FID_ID,"商品名")
    Call .DbSetFieldDataByID(TID_TEST1,FID_PRICE,"1200")
    rc = .DbAccessByID(BTR_INSERT,TID_TEST1,KID_ID)
    If rc <> 0 Then
        MsgBox CStr(rc)
    End If
End With
```

## システム・オプションについて

---

以下はシステム・オプションの設定の最初のページです。

設定項目についての説明は以下になります。

項目	概要
フィールド名入力時のIMEオン	フィールド情報設定ダイアログでフィールド名の入力時にIMEがローマ時かな入力に設定されます。
フィールド名英数字のUpper変換	DDFでは英数字はケースは無視される仕様です。このオプションを設定すると入力時にすべて大文字に変換します。
ゼロ・オフセット表示	フィールドやインデックスのグリッド表示で左に表示されるIDをゼロ・オフセットで表示します。バージョン1.xコンパチ関数等のパラメータでフィールドやインデックスは0オフセット表示なので、同関数を使う場合には0オフセット表示にしたほうが便利です。

DDF読み込時にIDのリナンバー	Pervasive Softwareがノベルだった時代に作成されたDDF作成用のDLLで作成されたDDFは内部IDがシリアルに設定されない場合があります。このオプションではDDF読み込み時にそのようなIDをシリアルな値で保持します。
DDF読み込時にウィンドウを開く	このチェックをはずすと既存のDDFをオープンした時には一旦テーブル・リストが作成されて、編集用のウィンドウは表示されません。テーブル数の多いDDFをオープンする場合には画面のリソースの消費を押さえることと表示のパフォーマンスが改善されます。
最大化する	チェックした場合にはDDFビルダー起動時にウィンドウを最大化します。
Btrieveデータ変換モード	チェックした場合には既存のBtrieveデータの形式をDDF定義の変更に合わせて変更します。

以下はシステム・オプション・ダイアログの出力タブです。



設定項目についての説明は以下です。

項目	概要
1 ページに印字するフォールド数	テーブルの印字で1 ページに印字されるフィールドの数を指定します。
フォーム生成	フォーム生成で生成されるフォームの形式を指定します。VB5用、VB4用は当ActiveX Controlで利用できるフォームです。VB2を選択した場合には16bitのVBMan for Btrieve/VBX用のフォームが出力されます。
文字列型シンボル	VBMan バージョン1.x関数でフィールド指定パラメータに利用できるシンボルを生成する場合はこの項目のチェックをはずします。VBManバージョン2.xからメソッドでの文字列型でテーブルやフィールドを設定する場合にはこの項目をチェックしてフィールドIDリストを出力します。



# Appendix

---

## A. Microsoft Visual Studio.NET 制約事項

---

### Refreshメソッドについて

Refreshメソッドは.NET frameworkのラップ・クラスによってオーバーライドされていると思われる、VBManのRefreshメソッドが呼び出されません。回避方法としては予約名になっていないReloadメソッドを追加しました。

### ImeModeについて

.NET frameworkではImeModeをデフォルトで全てのコントロールに追加します。VBManでもIMEModeを定義しているため、プロパティウィンドウにはImeMode/IMEModeと二つプロパティが並んで表示されます。VBManのプロパティは頭から3文字が大文字のIMEModeプロパティが該当します。.NET frameworkが付加するImeModeプロパティについては設定しても動作に関係しないことを確認しました。

### VBMEditコントロールのFormatStringについて

Microsoft Visual Studio .NETのVBFormatインターフェースが実装されていないと思われるため、EditコントロールのFormatStringプロパティは動作いたしません。アプリケーションコードで文字列フォーマットが必要になります。

## B. Active Server Pagesからの利用

---

概要・制約事項
---------

マイクロソフト社のWindows NT Option PackにはIIS(Internet Information Server)が含まれます。IISではVBScriptからActiveX Objectを利用することが出来る仕様になっており、VBManのデータ・ベース・コントロールを呼び出してBtrieveデータを操作することが可能です。VBScriptはActiveX Objectを呼び出せますが、ウィンドウを持つActiveX Controlは利用できない仕様ですからユーザー・インターフェースを持たないVBManのデータ・ベース・コントロールのみ利用できます。

また、言語の仕様がVisual Basicより簡略化されていることが原因でデータ・ベース・コントロールを使う上では制約があります。以下に制約事項をまとめました。

- ① プロパティはすべてコードで設定する必要がある。

- ② DbLoadDDFメソッドでDDFのロードをする必要がある。
- ③ AutoOpenプロパティは利用できないので、DbOpen/DbOpenAllでBtrieveファイルのオープンが必要になる。
- ④ 文字列型の配列をパラメータとして指定するDbGetRecordメソッド等は動作しない。(VBとパラメータの処理形式が異なるためと思われます)

## ASPスクリプトの作成手順

ASPスクリプトからVBManを参照可能とするためには、OBJECTタグでVBManを指定する必要があります。このタグは各.aspファイルの先頭で宣言します。また、global.asaファイルに指定することも可能です。以下はOBJECTタグでVBManを指定する例です。

```
<object RUNAT=Server id="VBMan" name="VBMan"
classid="clsid:3235CFF0-D891-11D1-BD2B-004026182472" border="0"
width="32" height="32">
</object>
```

次にBtrieveのオペレーション・コードを定義したファイルを.aspファイルにインクルードしてBTR\_INSERTのようなコンスタントを利用可能とします。VBManのサンプルに添付されるBtrieve.incファイルを以下のようにインクルード指定します。

```
<!-- #INCLUDE FILE="btrieve.inc" -->
```

DDFをロードするコードを記述します。DDFのロードはDbLoadDDFメソッドで実行可能です。このメソッドはDDFDirプロパティに指定されたディレクトリを参照しますので、メソッドを呼び出す前にDDFDirプロパティを指定します。以下はサンプル・コードです。

```
<% LANGUAGE="VBSCRIPT" %>
<% Dim rc %>
<% VBMan.DDFDir="\\OurServer\ddf" %>
<% VBMan.AllOpen=FALSE %>
<% rc = VBMan.DbLoadDDF %>
<% if rc <> 0 Then %>
  <p> DDFエラー <%= rc %>
<% End If%>
```

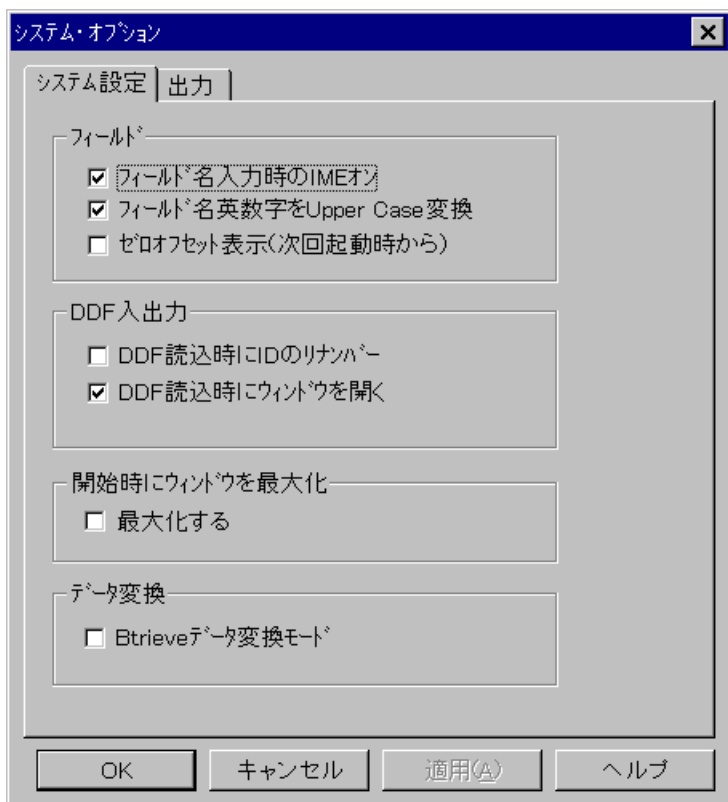
DDFをロードしたらBtrieveファイルをオープンします。DDFに定義されているすべてのテーブルを一度にオープンしたい場合はDbOpenAllメソッドが便利です。以下はサンプル・コードです。

```
<% rc = VBMan.DbOpenAll %>  
<% If rc <> 0 Then %>  
  <p> オープン・エラー <%= rc %>  
<% End If %>
```

Btrieveファイルがオープンできたら、データ・ベース・コントロールのメソッド呼び出しコードが記述できますので、お客様のアプリケーション・コードを記述してください。

処理の終了時には必ずDbCloseAllメソッドでBtrieveファイルをすべてクローズすることを忘れないでください。以下はサンプル・コードです。

```
<% rc = VBMan.DbCloseAll %>
```



## C. FAQ - よくあるご質問

---

この章では、VBManを使ったBtrieveアプリケーション・プログラミングやシステム・セットアップに共通の問題、疑問などの解説をします。

### **Pervasive.SQLで作成したカラムのサイズについて**

SQLで作成したカラムは”not null”制約をつけない場合はデータの先頭にヌル状態を保持するフラグ領域が1バイト確保されますのでレコード長等の計算時やBtrieveアプリで構造体の計算をする場合には注意が必要です。VBManではバージョン3からこのようなフィールドをサポートしません。

### **VBMan Controls for Btrieve の動作に必要なファイルは何か？**

以下のファイルをアプリケーションと一緒に配布してください。下記のファイル以外にはActian Zen の実行モジュールが必要になります。

VBMBT350.OCX  
DIBAPI32.DLL  
DDFIO350.DLL

### **vbmvt350.ocxを手動でレジストリ登録する**

regsvr32.exeを使ってCOM コンポーネントIを手動で登録することが可能です。管理者権限コマンドプロンプトで以下のコマンドを実行してください。

32bit 版OCX

```
cd “¥Program Files¥TechKnowledge¥VBMan controls for Btrieve 3.5¥bin”  
c:¥windows¥syswow64¥regsvr vbmvt350.ocx
```

64bit 版OCX

```
cd “¥Program Files¥TechKnowledge¥VBMan controls for Btrieve 3.5¥bin64”  
c:¥windows¥system32¥regsvr vbmvt350.ocx
```

### **regsvr32.exeで0x485のエラーとなる**

regsvr32.exeはモジュールのロードに失敗した場合、上記のようにC言語の16進形式でエラーを表示します。この値はWIN32 APIのGetLastErrorからの値です。0x485という値はモジュールが見つからない、ロードできないと言う意味に必要なDLLが存在しない場合のエラーです。vbmvt350ocxの登録でエラーになる場合はDIBAPI32.DLL,DDFIO350.DLL同じディレクトリに存在すること、Actian Zenが正常にインストールされていることをご確認ください。

## チェック・ボックスをクリックしても解除されない

VBManCheckBoxコントロールのDbValueTrueプロパティとDbValueFalseプロパティに同一の値が設定されています。異なる値をセットしてください。

## セグメント・キーの作成方法

複数項目で一つのキーとすることを、Btrieveではセグメント・キーといいます。VBManでセグメント・キーを作成する方法は以下の手順でおこないます。

1) プルダウンからインデックス、追加を選択。インデックス指定ダイアログでセグメントのチェック・ボックスをチェック

2) 複合キーにするフィールドを上記の手順を繰り返す。最後のフィールドを指定する時セグメントをチェックしない。セグメントがなしとする項目までで、ひとつの複合キーとなることに注意。

Btrieveではセグメント・キー全体でキー番号を一つだけ設定しますが、VBManにおいては管理方法が異なり、それぞれのフィールドにIDが割り振られることにご注意ください。データベース・アクセス関数DbAccessに指定するキー番号はセグメント・キーを構成する最初のフィールドIDから1引いた値を設定します。

以下は3つの項目でセグメント・キーを作成した場合のインデックス・グリッドの例です。

ID	名前	重複	修正可否	セグメント	ソート	ヌル
1	項目 1	不可	可	あり	昇順	不可
2	項目 2	不可	可	あり	昇順	不可
3	項目 3	不可	可	なし	昇順	不可
4						

以下はセグメント・キーでGET\_EQUALするVisual Basicでバージョン1.xコンパチブル関数を使ったサンプルです。フィールドITEM\_CDOE1, ITEM\_CODE2がセグメントで、Const文で設定している値はDDF定義のIDから1を引いたものを設定します。

```
Const ITEM_TABLE = 10
Const ITEM_CODE1 = 0
Const ITEM_CODE2 = 1
Const ITEM_KEY1 = 0
Dim rc As Integer, Code1$, Code2$

Code1$ = "100"
Code2$ = "ABC"
rc = DbSetFieldData( ITEM_TABLE, ITEM_CODE1, Code1$ )
```

```

rc = DbSetFieldData( ITEM_TABLE, ITEM_CODE2, Code2$ )
rc = DbAccess( BTR_GET_EQUAL, ITEM_TABLE, ITEM_KEY1 )
If rc <> 0 Then
    MsgBox "Btrieve Err" & Str$(rc)
End If

```

### DbListIndex指定に同じ名前のキーが表示される

同じフィールドで複数のセグメント・キーの先頭項目になっている場合、プロパティ・ダイアログで同じフィールド名がDbListIndexプロパティに選択肢に表示されます。これらを識別するためには、インデックス名(Named Index)をVBMan DDFビルダーで設定してください。設定の方法は名前をつけたいインデックスを選択して、（セグメント・キーの場合は先頭の項目）「インデックス」メニューから「インデックス名を追加」を選択してください。インデックス名を入力するダイアログが表示されますから、他のインデックスと識別可能な名前を設定してください。インデックス名はインデックス・グリッドに括弧で表示されます。

### DDFファイルのディレクトリの移動

VBMan DDFビルダーでデータベース定義し、フォームを定義した後、それらのファイルをまったく別のディレクトリに移動した場合、VBManデータベース・コントロールのDDFDirプロパティを移動したディレクトリに変更してください。変更せずに実行した場合の動作は保証できません。

### DDFディレクトリと別のディレクトリにBtrieveファイルをおきたい

VBMan DDFビルダーでテーブル名とBtrieveファイル名を設定するダイアログにおいて、Btrieveファイル名としてドライブとパスを指定することで可能です。ただし、ドライブ名を含めた場合は別環境で実行する場合（ドライブ名が異なる場合）にシステムやシステムの変更作業量が大きいと思われる為、お勧めしません。

### AutoIncrementがカウントしない

キーが設定されていない場合は整数型のフィールドと同じ扱いとなります。また、設定するキーは昇順のキーでなければなりません。

### VBManデータベースアクセス関数で集計処理の仕方

バージョン1.xコンパチブル関数を使った場合の集計処理のサンプル・コードを示します。

```

Const SALES_CODE 0
Const SALES_AMOUNT 3
Const SALES_CODE 0

```

```

Const SALES_AMOUNT 3
Const SALES_ITEM_KEY 0
Dim rc As Integer
Dim Total As Integer, Key$, Sales$

Total = 0
' 検索開始キーの値をバッファに設定
rc = DbSetFieldData( SALES_TABLE, SALES_ITEM, "AAA")
' BTRIEVEを呼び、最初に合致するレコードを得る
rc = DbAccess(BTR_GET_GREATER_OR_EQUAL, SALES_TABLE, SALES_ITEM_KEY )
Do While rc = 0
    Key$ = DbGetFieldData( SALES_TABLE, SALES_ITEM )
    If Key$ > "BBB" Then ' 集計範囲の検査
        Exit Do
    End If
' 集計処理
Sales$ = DbGetFieldData( SALES_TABLE, SALES_AMOUNT )
Total = Total + Val( Sales$ )
' BTRIEVEを呼び次のレコードを得る
rc = DbAccess(BTR_GET_NEXT,SALES_TABLE,SALES_ITEM_KEY)
Loop

If rc <> 9 And rc <> 0 Then
    MsgBox "Btrieve Err" & Str$(rc)
End If

```

### リクエスト環境でDDFOpenエラー(20)が起こる

Btrieveサーバーに接続できない状況が多いと思われます。クライアント・アプリケーションがBtrieveのリクエスター・モジュールにアクセスできない場合や、サーバーでBtrieveが実行されていない場合、サーバーと通信ができない場合などに発生します。

### Btrieveステータス2003について

ワークステーション・エンジンの設定が無いためにローカル・ディスクにあるBtrieveファイルにアクセスできない状況です。DDFファイル自体もBtrieveファイルなので、DDFビルダーでローカル・ディスクにあるDDFをオープンする場合にもこのステータスが表示されることがあります。

### 金額フィールドについて

VBManバージョン1.xではMoney型のフィールドについては自動的に金額フォーマットしましたが、VBMan Controls for BtrieveではFormatプロパティとFormatOptionプロパティの指定が必要です。前のバージョンと同じフォーマットをするにはプロパティの設定を以下のようにして



ください。Formatプロパティ、MaxLengthプロパティはフィールドのサイズによって変える必要があります。

```
VBManEdit1.Format = "###,###,###"  
VBManEdit1.FormatOption = 3  
VBManEdit1.NumericMask = True  
VBManEdit1.MaxLength = 11
```

### VBManボタンの押下をコードで実現したい

以下のコードで動作を確認しました。直接Clickイベントをよびだしても、Btrieveのオペレーションは実行されませんのでご注意ください。この方法はVisibleプロパティがTrueのボタンに対して有効です。ボタンをフォームの見えない位置に設定した場合でも動作します。

```
VBManBtn1.SetFocus  
SendKeys "{Enter}"
```

### 削除ボタンでカレント・レコードが無条件に消される

VBManBtnのDbOperationに削除を指定した場合、ボタンがクリックされた時点でのカレント・レコードが削除されてしまいます。フォームの入力とカレント・レコードが一致するかを検査するコードをVBManBtnのSetDataイベントに追加することで問題を解決できます。以下はサンプル・コードで、現在入力されている主キーでGET\_EQUALしてレコードの再ポジショニングをしています。

```
Sub VBManBtn1.SetData()  
Dim rc as Integer  
rc = DbTransferFromControl( TABLE_ID )  
rc = DbAccess(BTR_GET_EQUAL, TABLE_ID, PRIM_KEY)  
If rc = 4 Or rc = 9 Then  
    MsgBox "削除するデータが存在しません"  
    VBManBtn1.Abort = True  
    Exit Sub  
End If  
If rc <> 0 Then  
    MsgBox "Btrieve error " & str$(rc)  
    VBManBtn1.Abort = True  
    Exit Sub  
End If  
End Sub
```

## WaitLockの指定について

Btrieve for WindowsではWaitLockはサポートされていませんでしたが、Btrieve for Windows95/NTではマルチ・タスク環境に対応してWaitLockが可能になりました。

## Formatイベントのコード例

リスト・ボックス、コンボ・ボックス、拡張コンボ・ボックス、拡張リスト・ボックスのFormatイベントのコードはバージョン1.xではDbSetFormattedString,DbExSetFormattedString関数を使って設定していました。VBMan Controls for Btrieveではこれらの関数は使う必要がなく、パラメータとして渡される文字列を直接書き換えることでリストされる文字列を指定可能となりました。以前のバージョンとのコンパチビリティの為にこれらの関数も使用可能です。以下はVisual BasicにおけるFormatイベントのコード例です。

```
Sub VBManListBox1_Format( LineData As String )
Dim rc As Integer
Dim ItemName As String
' VBManリスト・ボックスではこの時点でカレント・レコードの
' フィールドからデータを得ることができる。
ItemName = VBManDB1.DbGetFieldData("商品","商品名")
' LineDataには商品のコードが入っているとす。
LineData = LineData & " - " & ItemName
End Sub
```

## DDFビルダーのVE1070メッセージ

データ変換モードのDDFビルダーで、Btrieveデータの形式を移行する場合、VE1070エラー・メッセージが表示されることがあります。このメッセージの括弧内はオペレーティング・システムからのエラー値です。(Win32 API GetLastErrorからの戻り値)エラー値5が表示される場合はBtrieveファイルが他のプロセスにより使用中のステータスです。Btrieveファイルを使っているプロセスを終了してください。

## D. バージョン履歴

---

以下はVBMan Controls for Btrieve ver 3.40で追加された機能です。

- ① Windows7に対応。
- ② Windows Vistaに対応。
- ③ Pervasive.SQL V10 データベース対応。

- ④ Microsoft Visual Studio .NET 2008に対応。
- ⑤ ATL version 9.0を採用。

以下はVBMan Controls for Btrieve ver 3.30で追加された機能です。

- ① Microsoft Visual Studio .NET 2005に対応。
- ② Pervasive.SQL V8 セキュアデータベース対応。
- ③ Pervasive.SQL V8.6 SP3対応。
- ④ ATL version 7.1を採用。

以下はVBMan Controls for Btrieve ver 3.20で追加された機能です。

- ① Microsoft .NET framework のWindows Forms対応
- ② Microsoft Visual Studio .NET 開発環境に対応。
- ③ BIT型,8byte Integer型データに対応。
- ④ ATL version 7.0を採用。
- ⑤ USB Sentinelのサポート。

以下は従来からのVBMan Controls for Btrieve の特徴です。

- ① ATLで作成された軽量コントロールです。MFC等サイズが大きいランタイムへの依存はありません。イントラネットでの配布が簡単になりました。
- ② アpartment・スレッド形式をサポート。サーバー・コンポーネントとして利用可能。
- ③ Pervasive.SQL 2000i (SP3)対応。
- ④ DDFに準拠したBtrieve呼び出しが簡単に記述できます。DDFに準拠したアプリケーションは他のアプリケーションとDDFを介してデータ交換が可能です。
- ⑤ DDFキャッシングを実現。複数コントロールが同じDDFを参照している場合はDDFをロードしないのでパフォーマンスが向上します。
- ⑥ 豊富なコントロール・プロパティにより、アプリケーションの開発工程を短縮できます。
- ⑦ dbMagicのデータ辞書からのDDF定義移行機能。

## エラー・メッセージ

この章では、VBManのエラー・メッセージを解説します。エラーメッセージは製品の改良のために予告なく追加、変更される場合があります。ご了承ください。

VE001 DDF読み込みエラー	DDFビルダーで生成したデータベース定義ファイルが読み込めません。
VE002 Btrieveを初期化できません	VBManがBtrieveの初期化を実行しましたが、エラーがBtrieveから戻されました。Btrieveの設定ファイル、インストールが完全か再度検査してください。
VE003 メモリーが足りません	実行時のメモリーが足りません。
VE004 DDFDirが指定されていません	VBManDBコントロールのDDFirプロパティが設定されていないため、オペレーションが可能ではない状態です。
VE005 DbTableが指定されていません	VBManコントロールのDbTableプロパティが設定されていないため、DbFieldプロパティを設定できない状態です。
VE006 DDF読み込みエラー	データベース定義ファイルの読み込み中にファイルの終わりとなりました。データベース定義ファイルが破壊されているか、VBManのバージョンが異なります。
VE007 Btrieveファイルの生成に失敗	データファイルが存在しなかったため、Btrieveデータファイルを生成しようとしたが、エラーがBtrieveから返されました。rc=xxとBtrieveのエラー・コードが表示されるので、Btrieveエラーを調べて対処してください。
VE008 Btrieveファイルのオープンに失敗(%d)	Btrieveデータファイルが破壊されています。Btrieveのユーティリティで復旧してください。
VE009 DDFファイルの形式が異なります	VBManのバージョンが異なります。または、データベース定義ファイルが破壊されています。
VE010 レコード数取得失敗(rc=%d)	レコード件数を取得できません。データベースが破壊されています。Btrieveのユーティリティで復旧が必要です。

VE011 データ型の不正	データ型がサポートされていません。使用しているデータ型が現在のVBManバージョンで使用可能かどうか確認してください。VBMan.DDFファイルが破壊されている可能性もあります。
VE013 イメージ・コントロールに適合しないデータ型です。	VBManの現在のバージョンでは.bmp,.ico,.wmf以外のファイル拡張子はイメージ・コントロールに表示できません。
VE014 このバージョンのVisual Basicでは動作しません。	VBMan Controls for BtrieveはVisual Basic 4.0J 32bit版で動作します。
VE015 イメージ・コントロールにビットマップを読み込めません。	.BMPファイルの形式が正しくありません。
VE018 VBManPixはフォームに最大個(10)存在します。	VBManPixはフォームに10個以上設定できません。
VE019 メタファイルが読めません。	データベースで指定されたメタファイルが存在しません。
VE020 アイコンファイルをオープン出来ません。	データベースで指定されたアイコンファイルが存在しません。
VE021 アイコンファイルのフォーマットが不正です。	指定されたアイコンファイルの形式が正しくありません。
VE023 アルダス形式のメタファイルを読み込めません。	指定されたメタファイルはアルダス形式と判断されましたが、その形式が正しくありません。
VE024 テーブルIDの指定が範囲外です。	データベース・アクセス関数に指定したテーブルのIDが範囲外です。
VE025 フィールドIDの指定が範囲外です。	データベース・アクセス関数に指定したフィールドのIDが範囲外です。
VE026 テンポラリ文字領域が確保できません。	DbGetFieldDataで戻るテンポラリの文字領域がありません。文字領域で使用しないものがあれば、開放してください。
VE027 インデックスIDの指定が範囲外です。	データベース・アクセス関数に指定したインデックスIDが範囲外です。
VE028 DDFオープン・エラー(nn)	DDFファイルをオープンできません。括弧内はBtrieveのステータス・コードです。この値をBtrieveマニュアル等を参考にして問題を解決してください。
VE029-031 DDF読み込みエラー	DDFファイルの形式が正しくありません。

VE035 キーIDの指定が範囲外です。	DDFで定義されていないキーのIDを指定しました。正しいキーIDを指定してください。
VE036 セグメント・キーの指定が不正です。	セグメント・キーの場合、セグメントの先頭のキーのIDを指定してください。
VE037 FindPercentageに失敗(%d)	Btrieve 6.XのFind Percentageオペレーションに失敗しました。括弧内はBtrieveからのステータス・コードです。Btrieveのファイル形式が5.xの場合にこのメッセージが表示される場合が多いです。ファイル形式を6.xに変換してください。
VE038 GetByPercentageに失敗(%d)	Btrieve 6.XのGet By Percentageオペレーションに失敗しました。括弧内はBtrieveからのステータス・コードです。Btrieveのファイル形式がバージョン5.xの場合にこのメッセージが表示されることが多いです。ファイル形式を6.xに変換してください。
VE039 Extended Opeartionエラー(%d)	Extended系Btrieveオペレーションが括弧内に表示されるエラー・コードを返しました。詳細はBtrieveプログラマーズ・マニュアル等を参照してプログラムを修正してください。
VE040 抽出するフィールド・リストが指定されていません。	Extended系Btrieveオペレーションにおいて、抽出するフィールドが指定されていません。
VE041 検索条件のフィールド指定には@が必要です。	Extended系オペレーションの検索条件の指定でフィールドIDが@プリフィックスになっていません。
VE042 比較演算子(=,<,>,<>,<=,>=)のいずれかを指定してください。	Extended系オペレーションの検索条件の指定で比較演算子の指定が不正です。正しい比較演算子を指定してください。
VE043 Extended系オペレーションに可変長データは指定できません。	Extended系オペレーションの抽出フィールドまたは検索条件にNote型、Lvar型のいずれかが指定されています。可変長型データをExtended系オペレーションに指定することはできません。
VE044 このオペレーション・コード(nn)は使用可能ではありません。	DbExAccess関数の第一パラメータに不正な値が設定されています。

<p>VE045 DbMaxRecordsが指定されていません。</p>	<p>拡張リスト・ボックスまたは拡張コンボ・ボックスのDbMaxRecordsプロパティに値0が指定されています。抽出するレコード件数を設定してください。</p>
<p>VE046データの指定がなく、&amp;   オペレータが指定されています。不正な検索条件文字列は(string)です。</p>	<p>Extended系コントロールまたは、メソッド/関数でDbConditionsプロパティまたはパラメータにおいて検索条件式の右辺がありませんでした。正しい検索条件式を設定してください。</p>
<p>VE050テーブルID(nn)のレコード長がDDF定義と一致していません。 Btrieve=xx,DDF=yy</p>	<p>Btrieveファイルをオープンして、レコードを入出力するメモリを割り振ろうとしましたが、Btrieveのレコード・サイズとDDFで定義されるレコードサイズが合致しませんでした。Btrieveファイルを生成しなおすか、既存のBtrieveファイルに合致するようにDDFを定義しなおしてください。BtrieveとDDFの定義が合致していることを確認するには既存のBtrieveファイルとDDFを定義して生成されたBtrieveファイルの両方にBUTILユーティリティ等でStatオペレーションを実行して、インデックスの定義、レコード長が同一であることを確認してください。DDFとBtrieveの定義が合致していない場合のVBManの動作は一切保証できません。</p>
<p>VE053センチネルエラーです。ステータス=(nn,nn)です。</p>	<p>センチネルを認識できませんでした。当マニュアルのセンチネルインストール手順に従って、ハードウェアの接続、ドライバーのインストール状況等をご確認ください。どうしても接続出来ない場合はカッコ内のステータスをサポートまでご連絡ください。</p>
<p>VE801オープン・モードの指定不正です。</p>	<p>VBManデータベース・コントロールのOpenModeプロパティの設定が正しくありません。</p>
<p>VE802 DbSourceまたはDbTableプロパティが不正です。</p>	<p>これらのプロパティを正しい値を指定してください。</p>

<p>VE803 Btrieveファイル(name)のオープンに失敗(nn)</p>	<p>BtrieveファイルnameをBtrieveステータスnnの理由でオープンすることができませんでした。ステータスの詳細をBtrieveプログラマーズ・マニュアル等で調査して問題を解決してください。</p>
<p>VE804インデックスの指定が不正です。</p>	<p>DbIndexまたはDbIndexNameプロパティの指定がDDF定義に存在しません。これらのプロパティの値を正しく設定してください。</p>
<p>VE805データ・ファイルnameをオープンできません(nn)</p>	<p>Btrieveデータ・ファイルnameをBtrieveステータスnnが原因でオープンすることができませんでした。ステータスの詳細をBtrieveプログラマーズ・マニュアル等で調査して問題を解決してください。</p>
<p>VE806 BtrieveファイルとDDFファイルの定義が合致していません。(Btrieve=mm,DDF=nn)</p>	<p>Btrieveファイルをオープンして、レコードを入出力するメモリを割り振ろうとしましたが、Btrieveのレコード・サイズとDDFで定義されるレコードサイズが合致しませんでした。Btrieveファイルを生成しなおすか、既存のBtrieveファイルに合致するようにDDFを定義しなおしてください。BtrieveとDDFの定義が合致していることを確認するには既存のBtrieveファイルとDDFを定義して生成されたBtrieveファイルの両方にBUTILユーティリティ等でStatオペレーションを実行して、インデックスの定義、レコード長が同一であることを確認してください。DDFとBtrieveの定義が合致していない場合のVBManの動作は一切保証できません。</p>
<p>VE807ボタンによるBtrieveオペレーションに失敗しました。(nn)</p>	<p>VBManボタン・コントロールをクリックされたので、Btrieveオペレーションを発行しましたが、Btrieveステータスnnが返されました。BtrieveステータスをBtrieveプログラマーズ・マニュアル等で調査してエラーを取り除いてください。</p>
<p>VE808 ParentのDispatchIDを取得できません。</p>	<p>VBManコントロールが置かれているフォーム等のOLE Dispatch IDが取得できません。このコントロール・コンテナはVBManを使用するには機能が不足しています。</p>



VE809 Parentディスパッチ・インターフェースを取得できません。	VBManコントロールが置かれているフォーム等のOLEディスパッチ・インターフェースが取得できません。このコントロール・コンテナはVBManを使用するには機能が不足しています。
VE810 ParentのName DispatchIDを取得できません。	VBManコントロールが置かれているフォーム等のNameプロパティのOLE Dispatch IDが取得できません。このコントロール・コンテナはVBManを使用するには機能が不足しています。
VE811 ParentのNameディスパッチ・インターフェースを取得できません。	VBManコントロールが置かれているフォーム等のNameプロパティのOLEディスパッチ・インターフェースが取得できません。このコントロール・コンテナはVBManを使用するには機能が不足しています。
VE812 DDFファイルクローズ・エラー(nn)	DDFファイルをクローズするときにBtrieveがステータスnnを返しました。このステータスをBtrieveプログラマーズ・マニュアル等で調査してエラーを取り除いてください。
VE813 Btrieveファイルnameのクローズ・エラー(nn)	Btrieveファイル名がnameであるBtrieveファイルをクローズするとき、Btrieveがnnのステータスを返しました。このステータスをBtrieveプログラマーズ・マニュアル等で調査してエラーを取り除いてください。
VE814コンテナからName, FormNameを取得できません。	コンテナ・アプリケーションの機能が不足しているので、VBManカスタム・コントロールを使うことができません。
VE815プロパティの値が不正です。	特定のコントロールのプロパティ値が不正です。
VE816パーセンテージ・オペレーションに失敗(nn)	スクロールバーの操作をするためにBtrieveのパーセンテージ・オペレーションを発行しましたが、Btrieveがステータスのnnを返しました。このステータスをBtrieveプログラマーズ・マニュアル等で調査してエラーを取り除いてください。

<p>VE817ゲット・パーセンテージ・オペレーションに失敗(nn)</p>	<p>スクロールバーの操作をするためにBtrieveのゲット・パーセンテージ・オペレーションを発行しましたが、Btrieveがステータスのnnを返しました。このステータスをBtrieveプログラマーズ・マニュアル等で調査してエラーを取り除いてください。</p>
<p>VE818 Extendedオペレーションで指定されたフィールド(field_name)が存在しません。</p>	<p>Extended系のオペレーションを実行する際にDbConditionsまたはDbListFieldsプロパティに指定されたフィールドfield_nameが見つかりませんでした。</p>
<p>VE819イメージ・ファイル(name)が存在しません。</p>	<p>VBManピクチャー・コントロールがイメージ・ファイルnameをオープンできませんでした。</p>

## VBManエラー・コード

VBMan Controls for Btrieveでは、メソッドや関数の戻り値にマイナスの値を返す場合があります。以下はこれらのVBManエラー・コードについての説明です。値(-16)以降はversion 3.50で新たに追加になったエラー・コードです。

エラー・シンボル	値	詳細
VBM_ERR_GENERIC	-1	一般的なエラー。これ以外のエラーに含めることができないもの。
VBM_ERR_ALREADY_OPEN	-2	Btrieveファイルがオープンしていない時に実行されるべきメソッド/関数がオープン中のファイルに対して実行された。
VBM_ERR_NOT_OPEN	-3	Btrieveファイルがオープンされている時に実行されるべきメソッド/関数がオープンされていないファイルに対して実行された。
VBM_ERR_INVALID_TABLE_NAME	-4	テーブル名がDDFDirプロパティで指定されるDDFに存在しない。
VBM_ERR_INVALID_FIELD_NAME	-5	フィールド名が存在しない。
VBM_ERR_INVALID_KEY	-6	インデックス名が存在しない。
VBM_ERR_INVALID_OPCODE	-7	Btrieveのオペレーション・コードとして指定できない値を設定した。
VBM_ERR_INVALID_EXCOND	-8	Extended系のBtrieveオペレーションを発行するメソッド/関数で検索条件の指定が不正である。
VBM_ERR_INVALID_EXTRACTOR	-9	Extended系のBtrieveオペレーションを発行するメソッド/関数で抽出するフィールドの指定が不正である。
VBM_ERR_LOCK_BIAS	-10	トランザクション関連のメソッド/関数などで、ロック値として不正な値を設定した。
VBM_ERR_CONTROL_NOT_FOUND	-11	DbAttachControl関数で最初のパラメータがただしくない。
VBM_ERR_NOT_ATTACHED	-12	DbAttachControl関数が呼び出されていない状態でバージョン1.xコンパチブル関数が呼び出された。
VBM_ERR_INVALID_FIELD_ID	-13	フィールドIDが正しくない。

VBM_ERR_INVALID_KEY_ID	-14	インデックスIDが正しくない。
VBM_ERR_INVALID_TABLE_ID	-15	テーブルIDが正しくない。
VBM_ERR_EXPAND_FILE_NAME	-16	DDFDirを短いファイル名に変換する際にエラーが発生しました。ファイル・システムの破損が考えられます。ScanDisk等でエラーが発生するファイル・システムを修復してください。
VBM_ERR_DATA_TYPE	-17	配列で変数を指定する仕様のメソッドにおいてパラメータのデータ型が不正です。
VBM_ERR_ARRAY_DIMS	-18	1次元配列が指定されたメソッドのパラメータにそれ以外の次元の配列が指定されました。
VBM_ERR_OUT_OF_RANGE	-19	データを受取る配列のサイズが不十分です。またはデータを配列で設定するメソッドの場合、配列のインデックス範囲がフィールドIDの範囲を越えています。
VBM_ERR_DDF_LOAD	-20	DDFがロードされていない状態でDDF情報が必要とされるメソッド、コントロールが利用されています。アプリケーションはDbLoadDDFメソッドで事前にDDFを読みこませる必要があります。
VBM_ERR_INVALID_DDF_DIR	-21	DbLoadDDFメソッドが呼び出されましたが、DDFDirプロパティに正しい値が設定されていませんでした。
VBM_ERR_ARRAY_ACCESS	-22	指定された配列をアドレスに変換するWin32 APIがエラーを返しました。Visual Basic等の言語では通常起こり得ないシステム・エラーです。言語やシステムの再インストールをお勧めします。
VBM_ERR_INVALID_ARRAY_TYPE	-23	配列で変数を指定する仕様のメソッドにおいてパラメータのデータ型が不正です。
VBM_ERR_INVALID_ARRAY_DIM	-24	1次元配列が指定されたメソッドのパラメータにそれ以外の次元の配列が指定されました。
VBM_ERR_SENT_MODULE_LOAD	-25	サポート用センチネル診断値です。センチネル用ドライバーが正常にインストールされていません。
VBM_ERR_SENT_MODULE_ENTRY	-26	サポート用センチネル診断値です。Sx32w.dllまたはsp32w.dllが破損している可能性があります。

VBM_ERR_SENT_QUERY	-28	サポート用センチネル診断値です。パラレルポート用センチネルが接続されていないと思われます。
VBM_ERR_SENT_MISSING	-29	サポート用センチネル診断値です。パラレルポート用センチネルが接続されていないと思われます。
VBM_ERR_SENT_OPEN	-30	サポート用センチネル診断値です。センチネルの初期化に失敗しています。
VBM_ERR_SENT_INIT	-31	サポート用センチネル診断値です。センチネルの初期化に失敗しています。
VBM_ERR_SENT_FIND_UNIT	-33	サポート用センチネル診断値です。USB センチネルのユニットが接続されていません。

## DDFビルダー・メッセージ

DDFビルダーが表示するメッセージについて解説します。

メッセージ	意味
VE1000 DDFディレクトリを指定してください	新規DDFの作成でDDFディレクトリが指定されていません。DDFディレクトリを正しく指定してください。
VE1001 DDF情報を正しく指定してください	新規DDFの作成でBtrieveファイル名が入力されていません。正しいBtrieveファイル名を指定してください。
VE1002 フィールド名を指定してください	フィールドの追加、修正でフィールド名が入力されていません。
VE1003 データサイズまたは小数点以下桁数が不正です	フィールドの追加、修正で設定したデータ型のサイズまたは小数点以下の設定が正しくありません。正確な値を設定してください。データ型の詳細をBtrieveマニュアルで参照されることをお勧めします。
VE1004 フィールドはこれ以上追加できません	現在のDDFビルダーで設定可能なフィールド数の上限に達しました。これ以上フィールドは追加することができません。
VE1006 すでに同名のフィールドが定義されています	フィールドの追加ですでにテーブルに存在するフィールド名を指定しました。フィールド名は他と識別するためにユニークな名前を設定する必要があります。
VE1007 すでに同名のテーブルが定義されています	新規テーブルの定義で指定したテーブル名はすでにDDFに存在します。テーブル名は他と識別するためにユニークな名前を設定する必要があります。
VE1008 フィールドが選択されていません	メニューからフィールド処理を選択しましたが、処理対象となるフィールドが選択されていません。マウスの左クリックで処理対象となるフィールドを選択してからメニューの選択を実行してください。
VE1009 テーブルが選択されていません	メニューからテーブルに関する処理が選択されましたが、処理対象となるテーブルが選択されていません。処理するテーブルのウィンドウのタイトルをクリックしてアクティブな状態にしてからメニューを選択してください。
VE1010 フィールド [%s] を削除しますか？	フィールド削除の確認です。削除しても良い場合はOKボタンをクリックしてください。
VE1011 フィールドが定義されていません	インデックスの追加をメニューから選択しましたが、まだフィールドがひとつも定義されていません。先にフィールドを定義してください。

VE1012 インデックスはこれ以上設定できません	インデックス定義の上限値に達しました。Btrieveの仕様により定義できるインデックス数には上限があります。
VE1013 フィールドが選択されていません	インデックスの追加・修正でフィールドが選択されていません。フィールドを選択してください。
VE1014 修正するインデックスが選択されていません	インデックスの修正をメニューから選択しましたが、修正対象となるインデックスが選択されていません。先にインデックスのグリッドから修正対象となるインデックスをマウスの左クリックで選択してください。
VE1015 インデックス[%s]を削除しますか?データ変換モードの場合で、セグメント・キーの場合はセグメント・キーを一括削除します。	インデックス削除の確認メッセージです。かぎ括弧で表示されたインデックスを削除してよければ、OKボタンをクリックしてください。データ変換モードではセグメント・キーの途中を残すことは意味が無いのでセグメント・キー全体を削除します。
VE1016 削除するインデックスが選択されていません	インデックスの削除をメニューから選択しましたが、削除対象となるインデックスが選択されていません。先にインデックスのグリッドから削除対象となるインデックスをマウスの左クリックで選択してください。
VE1017 DDFOpen APIエラー(%d)	既存のDDFファイルを修正するためにDDFをオープンしましたが、Btrieveから括弧内のステータスが返されました。括弧内の値をBtrieveのマニュアルなどで調査してください。多くの場合はBtrieveエンジンやサーバー環境の設定に問題があると思われます。
VE1018 DDFWriteFile APIエラー(%d)	FILE.DDFを書きこむ際にBtrieveが括弧内のステータスを返しました。括弧内の値をBtrieveのマニュアルなどで調査してください。多くの場合、Btrieveエンジンやサーバー環境の設定、ディスク容量などの問題思われます。
VE1019 DDFWriteField APIエラー[%s](%d)	FIELD.DDFを書きこむ際にBtrieveが括弧内のステータスを返しました。括弧内の値をBtrieveのマニュアルなどで調査してください。多くの場合、Btrieveエンジンやサーバー環境の設定、ディスク容量などの問題思われます。
VE1020 DDFWriteIndex APIエラー(%d)	INDEX.DDFを書きこむ際にBtrieveが括弧内のステータスを返しました。括弧内の値をBtrieveのマニュアルなどで調査してください。多くの場合、Btrieveエンジンやサーバー環境の設定、ディスク容量などの問題思われます。

VE1021 テーブルが定義されていません	DDFの保存を指定されましたが、テーブルがひとつも定義されていないので保存できません。
VE1022 インデックスの最後の項目はセグメントをオフにしてください	DDFの保存時にインデックスの最後の項目がセグメントがオンになったままで、セグメント・キーの終わりが設定されていません。インデックス定義の見直しが必要と思われます。
VE1023 データ型が選択されていません	フィールドの設定でデータ型が定義されていませんでした。データ型の選択は必須です。
VE1024 フィールドにはインデックスが作成されています。先にインデックスを削除してください	フィールドの削除が指定されましたが、対象となるフィールドにはインデックスが設定されています。フィールドを削除する前に関連するインデックスをすべて削除してからフィールドを削除してください。
VE1025 DDFファイルが存在しません	既存のDDFファイルのオープンが指定されましたが、FILE.DLL, FIELD.DDF, INDEX.DDFの3ファイルが同じディレクトリに存在しませんでした。DDFの存在するディレクトリを正しく指定してください。
VE1026 FILE.DDF読みエラー(%d)	既存のDDFファイルのオープンが指定されましたが、FILE.DDFの読み込み時にBtrieveが括弧内のステータスを返しました。括弧内の値をBtrieveマニュアルなどで調査してください。多くの場合、Btrieveエンジンの設定に問題があるか破損したDDFファイルを指定しているか、形式の正しくないDDFファイルを指定していると思われる。
VE1027 FIELD.DDF読みエラー(%d)	既存のDDFファイルのオープンが指定されましたが、FIELD.DDFの読み込み時にBtrieveが括弧内のステータスを返しました。括弧内の値をBtrieveマニュアルなどで調査してください。多くの場合、Btrieveエンジンの設定に問題があるか破損したDDFファイルを指定しているか、形式の正しくないDDFファイルを指定していると思われる。
VE1028 INDEX.DDF 読みエラー(%d)	既存のDDFファイルのオープンが指定されましたが、INDEX.DDFの読み込み時にBtrieveが括弧内のステータスを返しました。括弧内の値をBtrieveマニュアルなどで調査してください。多くの場合、Btrieveエンジンの設定に問題があるか破損したDDFファイルを指定しているか、形式の正しくないDDFファイルを指定していると思われる。



VE1031 vbman.iniへの書き込みに失敗しました	ウィンドウズがインストールしてあるディスクが満杯か破損している状況と思われます。ディスクの容量を確認できたら、ScanDisk等を実行して破損を回復してください。
VE1032 搬出ファイルを書き込みモードでオープンできません	搬出を指定したディスクの容量が不足しているかディスクが破損していると思われます。ディスクの容量を確認できたら、ScanDisk等を実行して破損を回復してください。
VE1033 テーブル[%s]の搬出が終了しました	搬出が正常に終了した場合の通知メッセージです。
VE1034 搬入ファイルを読み込みモードでオープンできません	搬入を指定したファイルかディスクが破損しているか、ファイルが存在しないと思われます。ファイルの存在が確認できたら、ScanDisk等を実行して破損を回復してください。
VE1035 テーブル[%s]の搬入が終了しました	搬入が正常に終了した場合の通知メッセージです。
VE1037 テーブル[%s]を削除しますか？	テーブルを削除する際の確認メッセージです。かぎ括弧で表示されるテーブルを削除する場合はOKボタンをクリックします。
VE1039 メモリが確保できません。	DDFビルダーが動作するのに十分なメモリが確保できませんでした。同時に起動しているアプリケーションがあれば終了させてください。DDFビルダー単独で起動しても同メッセージが表示される場合は、パソコンの実装メモリを増やすか、スワップを大きくしてください。
VE1040 BTRIEVE CREATEに失敗(%d)	Btrieveファイルの生成に失敗しました。失敗した理由は括弧内のBtrieveステータスで表示されていますので、Btrieveマニュアルなどで原因をお調べください。
VE1041 BTRIEVEファイル %s が存在します。上書きしますか？	Btrieveファイルの生成を指定しましたが、生成の対象となるファイルはすでに存在しています。既存のファイルが不要であれば、このメッセージにOKをクリックして新しいBtrieveファイルを上書きして生成します。
VE1042 ファイル %s はすでに存在します。上書きしますか？	フィールドIDファイルの出力で指定されたファイルはすでに存在しています。既存のファイルが不要であれば、このメッセージにOKをクリックして既存のファイルを上書きしてください。

VE1043 ファイルが書き込みモードでオープンできません。	フィールドIDファイルの出力で指定されたファイルを出力するモードでオープンすることが出来ませんでした。ディスクの容量が満杯になっているか、ディスクの破損が原因と思われます。ディスクの容量に問題が無ければ、ScanDiskなどでディスクの障害をとりのぞいてください。
VE1044 フィールドIDファイルの出力が終了しました	フィールドIDの出力が正常に終了したことを通知するメッセージです。
VE1045 DDFファイルを保存しますか	DDFファイルの保存確認メッセージです。
VE1046 書き込むフォーム %s のオープンに失敗しました	フォームの生成で指定されたファイルを書きこみのモードでオープンすることが出来ませんでした。ディスクの容量が満杯になっているか、ディスクの破損が原因と思われます。ディスクの容量に問題が無ければ、ScanDiskなどでディスクの障害をとりのぞいてください。
VE1047 新規または修正されたテーブル%sに対して整合性のないとおもわれるBTRIEVEファイル %s が存在します。OKをクリックした場合はBTRIEVEファイルを削除します	フィールド定義で合算されるレコード長と異なる既存のBtrieveファイル存在します。Controlではフィールド定義と合致しないBtrieveファイルを扱うことができないので、このメッセージにOKをクリックした場合は該当Btrieveファイルを削除します。
VE1048 BTRIEVEファイルの削除ができません。他のプログラムが使用中です	VE1047メッセージで削除を指定されたBtrieveファイルは他のアプリケーションなどでオープンされているため削除することが出来ませんでした。該当Btrieveファイルを削除したい場合はアプリケーションを終了させてから削除してください。
VE1049 テーブルにフィールドが定義されていません	フォームの生成が指定されましたが、指定されたテーブルにはフィールドがひとつも定義されていませんでした。フィールド定義、インデックス定義が完了したテーブルについてフォームの生成を実行してください。
VE1050 クリップボードの内容が不正です	クリップ・ボードにはVBMan DDFビルダーで設定した以外のデータが保持されているので、クリップ・ボードからのペーストが実行できませんでした。クリップ・ボードへのコピーからやり直してください。
VE1051 1ページに印字するフィールド数が不正です	システム・オプションの指定で1ページに印字するフィールド数の指定が0か空白になっています。0以外の値を指定してください。

VE1052 可変長フィールドを挿入することはできません。テーブルの最後に1フィールドのみの追加は可能です	フィールドの挿入を指定しましたが、設定したフィールドは可変長のデータ型です。可変長のフィールドはテーブルの最後に追加することだけが可能です。
VE1053 すでに可変長の項目が存在します	テーブルには可変長のデータ型を持つフィールドを複数定義することはできません。
VE1054 テーブルの最後の項目以外は可変長データ型に変更できません	フィールドの変更を指定しましたが、テーブルの最後の項目では無いため、可変長データ型のフィールドに変更することは出来ません。
VE1055 Named Indexは変更できません	インデックス名は追加と削除のみ可能な仕様です。
VE1057 テーブル[%s]にインデックスがありません。VBManからのDDF利用にはすくなくとも一つインデックスを作成してください	DDFの保存を指定しましたが、インデックスがひとつも定義されていません。VBManのコントロールを使う際にはインデックスが1個は必要なので、インデックスを定義してからDDFの保存をしてください。
VE1059 dbMagic情報ファイルの搬入に失敗しました	DbMagicデータ辞書ファイルの搬入に失敗したことを通知するメッセージです。
VE1060 dbMagicの数値形フィールドに不正なデータ長があります	DbMagicのデータ辞書の形式が不正です。今一度dbMagicでデータ辞書搬出をした結果をご確認ください。
VE1061 dbMagic情報ファイルのFILE行にDESC=が存在しません	DbMagicのデータ辞書の形式が不正です。今一度dbMagicでデータ辞書搬出をした結果をご確認ください。
VE1062 dbMagic情報ファイルのFILE行にNAME=(btrieveファイル名)の定義が存在しません	DbMagicのデータ辞書の形式が不正です。今一度dbMagicでデータ辞書搬出をした結果をご確認ください。
VE1063 インデックス名[%s]を削除しますか	インデックス名の削除確認メッセージです。かぎ括弧でかこまれたインデックス名を削除してもよければ、OKボタンをクリックしてください。
VE1064 インデックス名を設定するときはセグメント・キーの場合はセグメントの先頭にセットしてください	インデックス名の追加を指定する場合はセグメント・キーの先頭の項目に設定してください。セグメント・キーの先頭以外のメンバーを選択した状態でインデックス名の指定をメニューから選択しています。
VE1070 ファイルのコピーまたは移動に失敗しました。オペレーティングシステムからのエラー・コードは%dです。	データ移行モードでフォーマット変換の対象となるBtrieveデータ・ファイルを移動する際にオペレーティング・システムがエラーを返しました。エラー・コードはWin32 APIのGetLastErrorからの値です。Windows SDK等マイクロソフトの資料にエラー・コードの詳細が記載されています。

<p>VE1071 Btrieveファイルのフォーマット変換中にエラー(%d)</p>	<p>データ移行中にBtrieveが括弧内のエラーを返しました。括弧内の値をBtrieveマニュアルなどで調査してください。既存のデータにインデックスを追加する際に重複が発生する際には括弧内にステータス5が表示されます。</p>
<p>VE1072 ファイル変換につかたとおもわれるファイル %s が存在しません。削除して処理をすすめますか？</p>	<p>データ移行を開始する際に前回に変換に失敗したと思われる一時ファイルが残っていました。前回の変換で途中までの変換が残っていてそこからデータを復元できる場合もあります。一時ファイルが不要であればOKクリックして作業を進めてください。</p>
<p>VE1073 Btrieve ファイルがオープンできません。Btrieveのステータスは(%d)です。</p>	<p>データ移行の際に該当BtrieveファイルをオープンしたらBtrieveから括弧内のステータスが返されました。括弧内の値をBtrieveマニュアル等で調査してください。</p>
<p>VE1074 Btrieveデータ変換モードではフィールド、インデックスの修正はできません。変更したいフィールド、インデックスはいったん削除してから、追加してください。データを変換する必要がある場合は、システムオプションからデータ変換モードを解除してください。</p>	<p>データ変換モードでのフィールド変更に関する注意メッセージです。</p>
<p>VE1075 BtrieveのDropIndexオペレーションに失敗しました。Btrieveからのステータスは(%d)です</p>	<p>データ変換モードでインデックスの削除を指定されました。それに従ってBtrieveのキー削除を実行しましたが、Btrieveが括弧内のステータスを返してきました。括弧内の値をBtrieveマニュアル等で調査してください。</p>
<p>VE1076 BtrieveのCreateIndexオペレーションに失敗しました。Btrieveからのステータスは(%d)です</p>	<p>データ変換モードでインデックスの追加を指定されました。それに従ってBtrieveのキー追加を実行しましたが、Btrieveが括弧内のステータスを返してきました。括弧内の値をBtrieveマニュアル等で調査してください。</p>
<p>VE1078 フィールド数が多いのでフォームを作成してもVBで利用できません。(VB5/4では最大254コントロール/Form)</p>	<p>DDFのテーブルに定義されたフィールド数が254個を越えているので、フォームを生成してもVisual Basicで入力画面として利用することが出来ません。フィールド数の少ないテーブルを指定してください。</p>
<p>VE1079 dbMagic情報ファイルにFILE=の行が存在しません。</p>	<p>DbMagicデータ辞書ファイルの軽視kいが不正です。データ辞書ファイルの搬出結果を確認してください。</p>

## Btrieveステータス・コード

---

以下にBtrieveのステータス・コードと概要を記述します。このVbMan Controls for Btrieveのマニュアルが最初に記述された時点の情報であることに注意してください。最新版はActian Zen SDKドキュメントにあります。

2	I/O ERROR
3	File not opened
4	Key Value not found
5	Duplicate key value
6	Invalid key number
7	Different key number
8	Invalid Positioning
9	End of file
10	Modifiable Key value error
11	invalid file name
12	file not found
13	Extended file error
14	Pre-Image open error
15	Pre-Image I/O error
16	Expansion error
17	Close error
18	Disk Full
19	Unrecoverable error
20	Record Manager Inactive
21	Key Buffer Error
22	Key Buffer Error
23	Position Block Error
24	Page Size Error
25	Create I/O Error
26	Number of Keys
27	Key Position
28	Record Length
29	Key Length

30	Btrieve file name
31	Extended Error
32	Extended I/O error
34	Extend Name
35	Directory Error
37	Begin Transaction error
38	Transaction Control File
39	End/Abort Error
40	Transaction Max Files
41	Operation not allowed
42	Incomplete accelerated access
43	Invalid data record address
44	Null key path
45	Inconsistent Key flags
46	Access denied
47	Maximum open files
48	Invalid Alternate sequence definition
49	Key type error
50	Owner already set
51	Invalid Owner
52	Error Writing cache
53	Invalid Interface
54	Variable Page Unreadable
55	Autoincrement error
56	Incomplete index
57	Expanded Memory Error
58	Compression buffer too short
59	File Already Exists
60	Reject Count Reached
61	Work space too small
62	Incorrect descriptor
63	Invalid extended insert buffer
64	Filter limit reached

65	Incorrect field offset
74	Automatic transaction abort
75	Server Routing list too small
76	File server list too small
77	Wait lock error
78	Deadlock detected
80	Conflict
81	Lock error
82	Lost position
83	Read Outside Transaction
84	Record in use
85	File in use
86	File full
87	Handle full
88	Mode Error
90	Redirected Device table full
91	Server Error
92	Transaction table full
93	Incomplete lock type
94	Permission error
95	Session no longer valid
96	Communication environment error
97	Data message too small
98	Internal transaction error
1001	Invalid lock parameter
1002	Invalid memory parameter
1003	Insufficient memory for parameters specified
1004	Invalid page size parameter
1005	Invalid pre-image device parameter
1006	Invalid pre-image memory parameter
1007	Invalid file parameter
1008	Incorrect parameter
1009	Invalid transaction parameter

1010	Unable to access btrieve file for transaction recovery
1011	Invalid compression parameter
1012	Number of files in transaction should be between 1 to 18
1013	Task list is full
1014	File is open or transaction is active
1016	Already initialized
2001	Insufficient Memory
2002	Parameter Invalid or Out of Range
2003	No Local Access Allowed
2004	SPX is Not installed
2005	Wrong version of SPX Installed
2006	No Available SPX Connection
2007	Invalid Parameter



## VBMan Controls for Btrieve 調査依頼フォーム

---

以前はFAXで以下のような内容を記載してご質問くださいとお願いしていました。現在はメールで以下の概要を添えてご連絡ください。

日付	
会社名	
登録ユーザー名	
製品シリアル番号	
電話番号	
ファックス番号	
OSバージョン	
Btrieve/Actian Zenバージョン	
Visual Basic/Visual Studio.NETバージョン	
お問い合わせ内容、問題記述など、具体的に再現可能なようにご記入ください。	
添付資料	

VBMan Controls for Btrieve Version 3.50

第1版

2022年9月14日第1刷発行

版權・著作 株式会社テクノレッジ

Printed In Japan