Virtual CD API C Function Overview

The following list shows all functions implemented in the Virtual CD API C Interface for accessing, querying and changing virtual drives and CDs.

Table with all API functions

Function	**Description**
VCDApiEject	Eject a virtual CD from a virtual drive
VCDApiGetImagePropertiesLong	Query a numeric property from a virtual CD
VCDApiGetImagePropertiesString	Query a string property from a virtual CD
VCDApiGetLastError	Get the last error code
VCDApiGetMountDrive	Query the virtual drive letter in which a specific virtual CD is inserted
VCDApiGetMountedFileFromDrive	Query the virtual CD file name in a given virtual drive
VCDApiGetNumberOfVCDDevices	Get the number of existing virtual devices
VCDApiGetVCDDriveLetters	Get the currently existing virtual drive letters
VCDApiInsert	Insert a virtual CD
VCDApiIsProperlyInstalled	Test whether the Virtual CD API files are correctly installed
VCDApiIsVCDDriveLocked	Query whether a virtual drive is locked
VCDApiLockVCDDrive	Lock a virtual drive
VCDApiSetDebug	Activate additional debug messages
VCDApiSetDriveLetters	Add/remove virtual drive letters
VCDApiSetImagePropertiesLong	Set a numeric property for a virtual CD
VCDApiSetImagePropertiesString	Set a string property for a virtual CD
VCDApiUnLockVCDDrive	Unlock a virtual drive

VCDApiEject

Eject a virtual CD from a virtual drive.

```
DWORD VCDApiEject (
LPCSTR lpszFileName );

DWORD VCDApiEject (
LPCSTR lpszDrive );
```

Parameters

lpszFileName

Name of the virtual CD image file to be ejected. The file name must include the full path and the extension of the virtual CD.

lpszDrive

Drive letter of the virtual drive from which the CD is to be ejected.

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

There are two ways to eject a virtual CD:

1) By specifying a virtual drive: This method ejects the virtual CD from the specified drive. Only the drive letter is required, not the colon (e.g., "Z" for virtual drive Z:).

2) By specifying a virtual CD: If you do not know where a given virtual CD is inserted, the function can be called giving the full filename of the virtual CD. The function then checks which virtual drive the virtual CD is in and ejects it.

Example

```
// Eject a virtual CD from a virtual drive
DWORD dwRC = VCDApiEject("c:\\virtualCDs\\test.vc4");
...


// Eject CD from virtual drive X:
DWORD dwRC = VCDApiEject("X");
```

See Also

VCDApiInsert

VCDApiGetImagePropertiesLong

Query a numeric property of a virtual CD.

```
DWORD VCDApiGetImagePropertiesLong (
LPCSTR lpszFileName,
short iProperty );
```

Parameters

lpszFileName
Name of the virtual CD image file for which the query is valid. The file name must include the full path and the extension of the virtual CD.

iProperty
Type of property to be returned.

Return value

Numeric value based on the requested property type; 0 if the requested value is not assigned to the virtual CD.

Remarks

This function lets you query information assigned as a numeric value to a virtual CD. Possible property types for this function are:

- VCD_PROPERTY_HOTKEY
- VCD_PROPERTY_EJECTONEND

Example

```
// Query the currently assigned hotkey
DWORD dwHotKey = VCDApiGetImagePropertiesLong("c:\\virtualCDs\\test.vc4",
                                               VCD_PROPERTY_HOTKEY);
...

// Query whether the virtual CD is to be ejected when the current session ends
BOOL bEject;
bEject = (BOOL) VCDApiGetImagePropertiesLong("c:\\virtualCDs\\test.vc4",
                                             VCD_PROPERTY_EJECTONEND);
```

See Also

VCDApiGetImagePropertiesString, VCDApiSetImagePropertiesLong, VCDApiSetImagePropertiesString, Property types of a virtual CD

VCDApiGetImagePropertiesString

Query a string property of a virtual CD.

```
DWORD VCDApiGetImagePropertiesString (
LPCSTR lpszFileName,
short iProperty,
LPSTR lpszResult,
size_t iResultLen );
```

Parameters

lpszFileName

Name of the virtual CD image file for which the query is valid. The file name must include the full path and the extension of the virtual CD.

iProperty

Type of property to be returned

lpszResult

Buffer to store the result of the query (the requested property)

iResultLen

Length of the result buffer

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

This function lets you query information assigned as a string to a virtual CD. Possible property types for this function are:

- VCD_PROPERTY_DEFAULTDRIVE
- VCD_PROPERTY_DESCRIPTION
- VCD_PROPERTY_PROGRAM
- VCD_PROPERTY_WORKINGDIR
- VCD_PROPERTY_COMMENT

Example

```
// Query the default drive assigned to a virtual CD
char cResult[2];

DWORD dwRc = VCDApiGetImagePropertiesString("c:\\virtualCDs\\test.vc4",
                                             VCD_PROPERTY_DEFAULTDRIVE,
                                             cResult,
                                             2);

// Insert the virtual CD in the default drive
if(dwRC == VCD_ERROR_NONE)
        dwRC = VCDApiInsert(cResult[0],   "c:\\virtualCDs\\test.vc4", "");
```

See Also

VCDApiGetImagePropertiesLong, VCDApiSetImagePropertiesLong, VCDApiSetImagePropertiesString, Property types of a virtual CD

VCDApiGetLastError

Query the last API error code.

```
DWORD VCDApiGetLastError ();
```

Parameters

-

Return value

Last error code registered for a Virtual CD API function. For more information, see API error codes

Remarks

-

Example

```
DWORD dwRC = VCDApiGetLastError();
```

See Also

-

VCDApiGetMountDrive

Query the virtual drive in which a specified virtual CD is inserted.

```
char VCDApiGetMountDrive (
LPCSTR lpszFileName );
```

Parameters

lpszFileName

Name of the virtual CD image file for which the virtual drive letter is queried. The file name must include the full path and the extension of the virtual CD.

Return value

Virtual drive letter where the virtual CD is inserted. If the virtual CD is not inserted in any virtual drive, the return value is 0.

Remarks

This function lets you query the drive letter of the virtual drive in which a given virtual CD is inserted.

Example

```
// check whether the virtual CD is still inserted in a virtual drive
char cDrive = VCDApiGetMountDrive("c:\\virtualCDs\\test.vc4");
```

See Also

VCDApiGetMountedFileFromDrive

VCDApiGetMountedFileFromDrive

Query the name of a virtual CD image which is inserted in a specified virtual drive.

```
DWORD VCDApiGetMountedFileFromDrive (
char cDrive,
LPSTR lpszResult,
size_t iResultLen );
```

Parameters

cDrive

Virtual drive letter to query for the virtual CD image file name.

lpszResult

Buffer to store the file name of the inserted virtual CD.

iResultLen

Length of the result buffer.

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

This function returns the name of the virtual CD which is currently inserted in the specified virtual drive. If there is no virtual CD inserted in the virtual drive, the result buffer is empty.

Example

```
// Query which virtual CD is currently inserted in virtual drive X:
char cResult[MAX_PATH];
DWORD dwRC = VCDApiGetMountedFileFromDrive('X',   cResult, MAX_PATH);
```

See Also

VCDApiGetMountDrive

VCDApiGetNumberOfVCDDevices

Query the number of existing virtual drives

```
DWORD VCDApiGetNumberOfVCDDevices ();
```

Parameters

-

Return value

Number of existing virtual drives.

Remarks

This function lets you get the real number of virtual drives.

Example

```
DWORD dwRC = VCDApiGetNumberOfVCDDevices ();
```

See Also

VCDApiSetDriveLetters, VCDApiGetVCDDriveLetters

VCDApiGetVCDDriveLetters

Query the existing virtual drive letters.

```
DWORD VCDApiGetVCDDriveLetters (
LPSTR lpszResult,
size_t iResultLen );
```

Parameters

lpszResult
Buffer to store the virtual drive letters.

iResultLen
Length of the result buffer.

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

This function stores the existing virtual drive letters in its result buffer. For example, if you have 3 virtual drives designated X:, Y: and Z:, the result is "XYZ".

Example

```
// query which virtual drive letters currently exist
char cResult[27];
DWORD dwRC = VCDApiGetMountedFileFromDrive(cResult, 27);
```

See Also

VCDApiSetDriveLetters

VCDApiInsert

Insert a virtual CD into a virtual drive.

```
DWORD VCDApiInsert (
char cDrive,
LPCSTR lpszFileName,
LPCSTR lpszPassword );
```

Parameters

cDrive
Virtual drive letter.

lpszFileName
Name of the virtual CD image file to be inserted. The file name must include the full path and the extension of the virtual CD.

lpszPassword
Password of the virtual CD

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

To insert a virtual CD into a virtual drive, the drive letter and the name of the virtual CD are needed. A virtual CD can be inserted only in one virtual drive. If the virtual CD is still inserted in virtual drive X: when the command is received to insert it in virtual drive Z:, it is automatically ejected from virtual drive X: at the same time it is inserted in Z:.

The password must be left blank if the virtual CD does not require a password. If a password-protected virtual CD is to be inserted and the password input field is left empty, a dialog opens prompting the user for the password.

NOTE: The password is case sensitive!

Example

```
// Insert a virtual CD in virtual drive X: if it isn’t currently inserted in any virtual drive
char cDrive = VCDApiGetMountDrive("c:\\virtualCDs\\test.vc4");

if(cDrive == 0)
     DWORD dwRC = VCDApiInsert(’X’, "c:\\virtualCDs\\test.vc4", "");
...

// Insert a virtual CD in virtual drive Z:. Password of the virtual CD is Test.
DWORD dwRC = VCDApiInsert(’Z’, "c:\\virtualCDs\\test.vc4", "Test");
```

See Also

VCDApiEject

VCDApiIsProperlyInstalled

Check whether the Virtual CD API files are properly installed and ready to use.

```
DWORD VCDApiIsProperlyInstalled ();
```

Parameters

-

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

The function can be used to verify whether the Virtual CD API is properly installed; for example, before other functions are used.

Example

```
// Go on if the Virtual CD API is OK.
DWORD dwRC = VCDApiIsProperlyInstalled ();

if(dwRC == VCD_ERROR_NONE)
{
      ...
}
```

See Also

-

VCDApiIsVCDDriveLocked

Verify whether a virtual drive is locked.

```
BOOL VCDApiIsVCDDriveLocked (
TCHAR cDrive );
```

Parameters

cDrive

Virtual drive to be checked for locking.

Return value

TRUE if the drive is locked; otherwise, FALSE.

Remarks

This function verifies whether a virtual drive is locked.

Example

```
// Test whether virtual drive Y: is locked and, if so, unlock it
if(VCDApiIsVCDDriveLocked('Y'))
      DWORD dwRC = VCDApiUnLockVCDDrive('Y');
```

See Also

VCDApiLockVCDDrive, VCDApiUnLockVCDDrive

VCDApiLockVCDDrive

Lock a virtual drive.

```
void VCDApiLockVCDDrive (
TCHAR cDrive );
```

Parameters

cDrive

Virtual drive to be locked.

Return value

-

Remarks

Locking a virtual drive is useful if you wish to block any action affecting this drive. For example, it is not possible to eject a virtual CD from a locked virtual drive using the Explorer.

NOTE: Before a locked virtual drive can be used again, it must be unlocked!

Example

```
// Lock virtual drive Y:
DWORD dwRc = VCDApiLockVCDDrive('Y');
```

See Also

VCDApiIsVCDDriveLocked, VCDApiUnLockVCDDrive

VCDApiSetDebug

Activate additional debug messages.

```
void VCDApiSetDebug (
BOOL bDebug );
```

Parameters

bDebug

TRUE to activate additional debug messages; FALSE to deactivate them.

Return value

-

Remarks

If additional debug messages are activated, the functions open dialog boxes showing additional information when an error occurs.

NOTE: This function should be used only during development!

Example

```
// Activate additional debug information during development
VCDApiSetDebug(TRUE);
```

See Also

-

VCDApiSetDriveLetters

Add or remove virtual drives.

```
DWORD VCDApiSetDriveLetters (
LPCSTR lpszDriveLetters );
```

Parameters

lpszDriveLetters

Buffer with virtual drive letter to set.

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

To change the number of virtual drives and their drive letters, define a buffer with the drive letters in question. For example, if you want to define X:, Y: and Z: as virtual drives, call the function as follows: VCDApiSetDriveLetters("XYZ").

NOTE: Use of this function should be thought through carefully. Changing virtual drives too often, for example, can create problems in some systems. Furthermore, the following factors can cause difficulties when using this function:

- The amount of time it takes to change virtual drives: Some operating systems might take as long as several minutes to change virtual drives. Windows XP seems to be the fastest OS when it comes to changing virtual drives. Windows 2000 takes much longer. There is no way to predict how long it will take to change the virtual drives; it is a good idea to inform users that the action might take a while.
- Computer restart required to activate changes: Depending on the operating system used with the Virtual CD API, it might be necessary to restart the computer before newly set up virtual drives are available. Windows NT4, for example, does not support plug-and-play drivers and must be restarted for every change of virtual drives. Depending on internal system factors, however, rebooting might be necessary with other operating systems as well.
- Multiple applications using the Virtual CD API: If there are multiple applications installed using the Virtual CD API, they might use different drive letters. To be sure that an application gets all the virtual drive letters needed, check which drive letters exist.

NOTE: When this function is used only to change the letters of the virtual drives (e.g., LMN to XYZ), but not the number of drives, it is much faster than when removing or adding virtual drives.

Example

```
// Set virtual drives to X: Y: and Z:
DWORD dwRC = VCDApiSetDriveLetters("XYZ");

If(dwRc != VCD_ERROR_NONE)
{
      If(dwRC == VCD_ERROR_REBOOTREQUIRED)
      {
        // restart computer
        …
      }
      else
      {
        // additional error handling
        …
      }
}
```

See Also

VCDApiGetNumberOfVCDDevices, VCDApiGetVCDDriveLetters

VCDApiSetImagePropertiesLong

Set a numeric property for a virtual CD.

```
DWORD VCDApiSetImagePropertiesLong (
LPCSTR lpszFileName,
short iProperty,
long lNewVal );
```

Parameters

lpszFileName
Name of the virtual CD image file for which the property is to be set. The file name must include the full path and the extension of the virtual CD.

iProperty
Type of property to be set.

lNewVal
New value for the property.

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

This function lets you set a property assigned as a numeric value for a virtual CD. Possible property types for this function are:

- VCD_PROPERTY_HOTKEY
- VCD_PROPERTY_EJECTONEND

Example

```
// Change the Eject on End property so that the virtual CD is ejected on session end
DWORD dwRC = VCDApiSetImagePropertyLong("c:\\virtualCDs\\test.vc4",
                                        VCD_PROPERTY_EJECTONEND,
                                        1);
```

See Also

VCDApiGetImagePropertiesLong, VCDApiGetImagePropertiesString, VCDApiSetImagePropertiesString, Property types of a virtual CD

VCDApiSetImagePropertiesString

Set a string property for a virtual CD.

```
DWORD VCDApiSetImagePropertiesString (
LPCSTR lpszFileName,
short iProperty,
LPCSTR lpszNewValue );
```

Parameters

lpszFileName

Name of the virtual CD image file for which the property is to be set. The file name must include the full path and the extension of the virtual CD.

iProperty

Type of property to be set.

lpszNewVal

New value for the property.

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

This function lets you set a property assigned as a string for a virtual CD. Possible property types for this function are:

- VCD_PROPERTY_DEFAULTDRIVE
- VCD_PROPERTY_DESCRIPTION
- VCD_PROPERTY_PROGRAM
- VCD_PROPERTY_WORKINGDIR
- VCD_PROPERTY_COMMENT

Example

```
// Change the 'program' property for a virtual CD
DWORD dwRC = VCDApiSetImagePropertyLong("c:\\virtualCDs\\test.vc4",
                                     VCD_PROPERTY_PROGRAM,
                                     "explorer.exe");
```

See Also

VCDApiGetImagePropertiesLong, VCDApiGetImagePropertiesString, VCDApiSetImagePropertiesLong, Property types of a virtual CD

VCDApiUnLockVCDDrive

Unlock a virtual drive.

```
void VCDApiUnLockVCDDrive (
TCHAR cDrive );
```

Parameters

cDrive

Virtual drive to be unlocked.

Return value

-

Remarks

This function unlocks a virtual drive. This must be done before a drive that had been locked can be used again.

Example

```
// Check whether virtual drive Y: is locked and, if so, unlock it
if(VCDApiIsVCDDriveLocked('Y'))
      DWORD dwRC = VCDApiUnLockVCDDrive('Y');
```

See Also

VCDApiIsVCDDriveLocked, VCDApiLockVCDDrive

Virtual CD IApi COM Interface Overview

The following list shows all functions which can be used for accessing, querying and changing virtual drives and CDs once the Virtual CD IAPI Interface has been initialized.

Table of all API functions:

Function	**Description**
VCDEject	Eject a virtual CD from a virtual drive
VCDGetImagePropertiesLong	Query a numeric property from a virtual CD
VCDGetImagePropertiesString	Query a string property from a virtual CD
VCDGetLastError	Get the last error code
VCDGetMountDrive	Query the virtual drive letter in which a specific virtual CD is inserted
VCDGetMountedFileFromDrive	Query the virtual CD file name in a given virtual drive
VCDGetNumberOfVCDDevices	Get the number of existing virtual devices
VCDGetVCDDriveLetters	Get the currently existing virtual drive letters
VCDInsert	Insert a virtual CD
VCDIsProperlyInstalled	Test whether the Virtual CD API files are correctly installed
VCDIsVCDDriveLocked	Query whether a virtual drive is locked
VCDLockVCDDrive	Lock a virtual drive
VCDSetDebug	Activate additional debug messages
VCDSetDriveLetters	Add/remove virtual drive letters
VCDSetImagePropertiesLong	Set a numeric property for a virtual CD
VCDSetImagePropertiesString	Set a string property for a virtual CD
VCDUnLockVCDDrive	Unlock a virtual drive

Initializing the IApi COM Interface

Before you can initialize the IApi COM Interface must be registered. Normally this is done by the Virtual CD installation process. Once installed the Interface can be used as follows:

```
' Init the VCD API object
DIM IApi
Set IApi = CreateObject("VC5API.Api")
```

VCDEject

Eject a virtual CD from a virtual drive.

```
Function VCDEject (
ByVal FileName As String
) As Long

Function VCDEject (
ByVal Drive As String
) As Long
```

Parameters

FileName

Name of the virtual CD image file to be ejected. The file name must include the full path and the extension of the virtual CD.

Drive

Drive letter of the virtual drive from which the CD is to be ejected.

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

There are two ways to eject a virtual CD:

1) By specifying a virtual drive: This method ejects the virtual CD from the specified drive. Only the drive letter is required, not the colon (e.g., "Z" for virtual drive Z:).
2) By specifying a virtual CD: If you do not know where a given virtual CD is inserted, the function can be called giving the full filename of the virtual CD. The function then checks which virtual drive the virtual CD is in and ejects it.

Example

```
 ‘ Eject a virtual CD from a virtual drive
iRC = IApi.VCDEject("c:\virtualCDs\test.vc4");
…


‘ Eject CD from virtual drive X:
iRC = IApi.VCDEject("X");
...
```

See Also

VCDInsert

VCDGetImagePropertiesLong

Query a numeric property of a virtual CD.

```
Function VCDGetImagePropertiesLong (
ByVal FileName As String,
ByVal Property As Integer
) As Long
```

Parameters

FileName

Name of the virtual CD image file for which the query is valid. The file name must include the full path and the extension of the virtual CD.

Property

Type of property to be returned.

Return value

Numeric value based on the requested property type; 0 if the requested value is not assigned to the virtual CD.

Remarks

This function lets you query information assigned as a numeric value to a virtual CD. Possible property types for this function are:

- VCD_PROPERTY_HOTKEY
- VCD_PROPERTY_EJECTONEND

Example

```
 ‘ Query the currently assigned hotkey
iHotKey = IApi.VCDGetImagePropertiesLong("c:\virtualCDs\test.vc4",
                                          VCD_PROPERTY_HOTKEY);
...

‘ Query whether the virtual CD is to be ejected when the current session ends
iEjectOnEnd = IApi.VCDGetImagePropertiesLong("c:\virtualCDs\test.vc4",
                                              VCD_PROPERTY_EJECTONEND);
```

See Also

VCDGetImagePropertiesString, VCDSetImagePropertiesLong, VCDSetImagePropertiesString.
Property types of a virtual CD

VCDGetImagePropertiesString

Query a string property of a virtual CD.

```
Function VCDGetImagePropertiesString (
ByVal FileName As String,
ByVal Property As Integer
) As String
```

Parameters

FileName

Name of the virtual CD image file for which the query is valid. The file name must include the full path and the extension of the virtual CD.

Property

Type of property to be returned.

Return value

String value based on the requested property type.

Remarks

This function lets you query information assigned as a string to a virtual CD. Possible property types for this function are:

- VCD_PROPERTY_DEFAULTDRIVE
- VCD_PROPERTY_DESCRIPTION
- VCD_PROPERTY_PROGRAM
- VCD_PROPERTY_WORKINGDIR
- VCD_PROPERTY_COMMENT

Example

```
strDefaultDrive = IApi.VCDGetImagePropertiesString("c:\virtualCDs\test.vc4",
                                                   VCD_PROPERTY_DEFAULTDRIVE)

' Get the last error code
iRC = IApi.VCDGetLastError()

‘ Insert the virtual CD in the default drive
If iRC = VCD_ERROR_NONE Then
        iRC = IApi.VCDInsert(strDefaultDrive, "c:\virtualCDs\test.vc4", "")
        …
End If
```

See Also

VCDGetImagePropertiesLong, VCDSetImagePropertiesLong, VCDSetImagePropertiesString, Property types of a virtual CD

VCDGetLastError

Query the last API error code.

Function VCDGetLastError () As Long

Parameters

-

Return value

Last error code registered for a Virtual CD API function. For more information, see API error codes

Remarks

-

Example

```
' Get the last error code
iRC = IApi.VCDGetLastError()
```

See Also

-

VCDGetMountDrive

Query the virtual drive in which a specified virtual CD is inserted.

```
Function VCDGetMountDrive (
ByVal FileName As String
) As String
```

Parameters

FileName

Name of the virtual CD image file for which the virtual drive letter is queried. The file name must include the full path and the extension of the virtual CD.

Return value

Virtual drive letter where the virtual CD is inserted. If the virtual CD is not inserted in any virtual drive, the return value is empty.

Remarks

This function lets you query the drive letter of the virtual drive in which a given virtual CD is inserted.

Example

```
 ‘ check if the virtual CD is still inserted into a virtual drive
strDrive = IApi.VCDGetMountDrive("c:\virtualCDs\test.vc4")
```

See Also

VCDGetMountedFileFromDrive

VCDGetMountedFileFromDrive

Query the name of a virtual CD image which is inserted in a specified virtual drive.

```
Function VCDGetMountedFileFromDrive (
ByVal DriveLetter As String
) As String
```

Parameters

DriveLetter
Virtual drive letter to query for the virtual CD image file name.

Return value

Name of the virtual CD image, or empty if there is no virtual CD inserted.

Remarks

This function returns the name of the virtual CD which is currently inserted in the specified virtual drive. If there is no virtual CD inserted in the virtual drive, the result is empty.

Example

```
 ‘ Query which virtual CD is currently inserted in virtual drive X:
strFileName = IApi.VCDGetMountedFileFromDrive("X")
```

See Also

VCDGetMountDrive

VCDGetNumberOfVCDDevices

Query the number of existing virtual drives

Function VCDGetNumberOfVCDDevices () As Long

Parameters

-

Return value

Number of existing virtual drives.

Remarks

This function lets you get the real number of virtual drives.

Example

```
DIM IApi
DIM iDevices

' Init the VCD API object
Set IApi = CreateObject("VC5API.Api")

' query the number of virtual devices
iDevices = IApi.VCDGetNumberOfVCDDevices()
```

See Also

VCDGetVCDDriveLetters, VCDSetDriveLetters

VCDGetVCDDriveLetters

Query the existing virtual drive letters.

```
Function VCDGetVCDDriveLetters() As String
```

Parameters

-

Return value

Currently existing virtual drive letters.

Remarks

This function stores the existing virtual drive letters in its result buffer. For example, if you have 3 virtual drives designated X:, Y: and Z:, the result is "XYZ".

Example

```
 ‘ query which virtual drive letters currently exist
strDrives = IApi.VCDGetMountedFileFromDrive()
```

See Also

VCDSetDriveLetters

VCDInsert

Insert a virtual CD into a virtual drive.

```
Function VCDInsert (
ByVal Drive As String,
ByVal FileName As String,
ByVal Password As String
) As Long
```

Parameters

Drive

Virtual drive letter.

FileName

Name of the virtual CD image file to be inserted. The file name must include the full path and the extension of the virtual CD.

Password

Password of the virtual CD

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

To insert a virtual CD into a virtual drive, the drive letter and the name of the virtual CD are needed. A virtual CD can be inserted only in one virtual drive. If the virtual CD is still inserted in virtual drive X: when the command is received to insert it in virtual drive Z:, it is automatically ejected from virtual drive X: at the same time it is inserted in Z:.

The password must be left blank if the virtual CD does not require a password. If a password-protected virtual CD is to be inserted and the password input field is left empty, a dialog opens prompting the user for the password.

NOTE: The password is case sensitive!

Example

```
 ‘Insert a virtual CD in virtual drive X: if it isn’t currently inserted in any virtual drive
strDrive = IApi.VCDGetMountDrive("c:\virtualCDs\test.vc4")

' Insert the CD if it is not currently in a drive
If Len(strDrive) = 0 then
        iRC = IApi.VCDInsert("", "c:\virtualCDs\test.vc4", "")

...

‘ Insert a virtual CD in virtual drive Z:. Password for the virtual CD is Test.
iRC = IApi.VCDInsert("Z", "c:\virtualCDs\test.vc4", "Test")
```

See Also

VCDEject

VCDIsProperlyInstalled

Check whether the Virtual CD API files are properly installed and ready to use.

Function VCDIsProperlyInstalled() As Long

Parameters

-

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

The function can be used to verify whether the Virtual CD API is properly installed; for example, before other functions are used.

Example

```
 ‘ Go on if the Virtual CD API is OK.
iRC = IApi.VCDIsProperlyInstalled()

If iRC = 0 Then
      …
End If
```

See Also

-

VCDIsVCDDriveLocked

Verify whether a virtual drive is locked.

```
Function VCDIsVCDDriveLocked (
ByVal Drive As String
) As Long
```

Parameters

Drive
Virtual drive to be checked for locking.

Return value

1 if the drive is locked; otherwise, 0.

Remarks

This function verifies whether a virtual drive is locked.

Example

```
 ‘ Test whether virtual drive Y: is locked and, if so, unlock it
iRC = IApi.VCDIsVCDDriveLocked("Y")

if iRC = 1 Then
        IApi.VCDUnLockVCDDrive("Y")
End If
```

See Also

VCDLockVCDDrive, VCDUnLockVCDDrive

VCDLockVCDDrive

Lock a virtual drive.

```
Sub VCDLockVCDDrive (
ByVal Drive As String )
```

Parameters

Drive

Virtual drive to be locked.

Return value

-

Remarks

Locking a virtual drive is useful if you wish to block any action affecting this drive. For example, it is not possible to eject a virtual CD from a locked virtual drive using the Explorer.

NOTE: Before a locked virtual drive can be used again, it must be unlocked!

Example

```
 ‘ Lock virtual drive Y:
IApi. VCDLockVCDDrive("Y")
```

See Also

VCDIsVCDDriveLocked, VCDUnLockVCDDrive

VCDSetDebug

Activate additional debug messages.

```
Sub VCDSetDebug (
ByVal bDebugTrace As Long )
```

Parameters

Debug

1 to activate additional debug messages; 0 to deactivate them.

Return value

-

Remarks

If additional debug messages are activated, the functions open dialog boxes showing additional information when an error occurs.

NOTE: This function should be used only during development!

Example

```
‘ Activate addition debug information during development
IApi.VCDSetDebug(1);
```

See Also

-

VCDSetDriveLetters

Add or remove virtual drives.

```
Function VCDSetDriveLetters (
ByVal DriveLetters As String
) As Long
```

Parameters

DriveLetters

Buffer with virtual drive letter to set.

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

To change the number of virtual drives and their drive letters, define a buffer with the drive letters in question. For example, if you want to define X:, Y: and Z: as virtual drives, call the function as follows: VCDApiSetDriveLetters("XYZ").

NOTE: Use of this function should be thought through carefully. Changing virtual drives too often, for example, can create problems in some systems. Furthermore, the following factors can cause difficulties when using this function:

- The amount of time it takes to change virtual drives: Some operating systems might take as long as several minutes to change virtual drives. Windows XP seems to be the fastest OS when it comes to changing virtual drives. Windows 2000 takes much longer. There is no way to predict how long it will take to change the virtual drives; it is a good idea to inform users that the action might take a while.
- Computer restart required to activate changes: Depending on the operating system used with the Virtual CD API, it might be necessary to restart the computer before newly set up virtual drives are available. Windows NT4, for example, does not support plug-and-play drivers and must be restarted for every change of virtual drives. Depending on internal system factors, however, rebooting might be necessary with other operating systems as well.
- Multiple applications using the Virtual CD API: If there are multiple applications installed using the Virtual CD API, they might use different drive letters. To be sure that an application gets all the virtual drive letters needed, check which drive letters exist.

Example

```
‘ Set virtual drives to X: Y: and Z:
iRC = IApi.VCDSetDriveLetters("XYZ");

If iRC <> VCD_ERROR_NONE Then
    If dwRC = VCD_ERROR_REBOOTREQUIRED Then
      ‘ restart computer
      …
    Else
      ‘ additional error handling
      …
    End If
End If
```

See Also

VCDGetNumberOfVCDDevices, VCDGetVCDDriveLetters

VCDSetImagePropertiesLong

Set a numeric property for a virtual CD.

```
Function VCDSetImagePropertiesLong (
ByVal FileName As String,
ByVal Property As Integer,
ByVal lNewVal As Long
) As Long
```

Parameters

FileName

Name of the virtual CD image file for which the property is to be set. The file name must include the full path and the extension of the virtual CD.

Property

Type of property to be set.

NewVal

New value for the property.

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

This function lets you set a property assigned as a numeric value for a virtual CD. Possible property types for this function are:

- VCD_PROPERTY_HOTKEY
- VCD_PROPERTY_EJECTONEND

Example

```
‘ Change the Eject on End property so that the virtual CD will be ejected on session end
iRC = lApi.VCDSetImagePropertyLong("c:\virtualCDs\test.vc4",
                                                VCD_PROPERTY_EJECTONEND,
                                                1);
```

See Also

VCDGetImagePropertiesLong, VCDGetImagePropertiesString, VCDSetImagePropertiesString, Property types of a virtual CD

VCDSetImagePropertiesString

Set a string property for a virtual CD.

```
Function VCDSetImagePropertiesString (
ByVal FileName As String,
ByVal Property As Integer,
ByVal newVal As String
) As Long
```

Parameters

FileName

Name of the virtual CD image file for which the property is to be set. The file name must include the full path and the extension of the virtual CD.

Property

Type of property to be set.

NewVal

New value for the property.

Return value

0 if the function is successful; otherwise, it is nonzero. For more information, see API error codes

Remarks

This function lets you set a property assigned as a string for a virtual CD. Possible property types for this function are:

- VCD_PROPERTY_DEFAULTDRIVE
- VCD_PROPERTY_DESCRIPTION
- VCD_PROPERTY_PROGRAM
- VCD_PROPERTY_WORKINGDIR
- VCD_PROPERTY_COMMENT

Example

```
‘ Change the 'program' property for a virtual CD
iRC = IApi.VCDSetImagePropertyLong("c:\virtualCDs\test.vc4",
                                   VCD_PROPERTY_PROGRAM,
                                   "explorer.exe");
```

See Also

VCDGetImagePropertiesLong, VCDGetImagePropertiesString, VCDSetImagePropertiesLong, Property types of a virtual CD

VCDUnLockVCDDrive

Unlock a virtual drive.

```
Sub VCDUnLockVCDDrive (
ByVal Drive As String )
```

Parameters

Drive

Virtual drive to be unlocked.

Return value

-

Remarks

This function unlocks a virtual drive. This must be done before a drive that had been locked can be used again.

Example

```
‘ Check whether virtual drive Y: is locked and, if so, unlock it
iRC = IApi.VCDIsVCDDriveLocked("Y")

if iRC = 1 Then
      IApi.VCDUnLockVCDDrive("Y")
End If
```

See Also

VCDIsVCDDriveLocked, VCDLockVCDDrive

Virtual CD API

The Virtual CD API lets you integrate the powerful virtual CD functionality into other programs. Its architecture is independent from Virtual CD v5.x.

With Virtual CD API, you can

Ø eject and insert virtual CDs.

Ø protect your virtual CDs through password control.

Ø add and remove virtual drives as needed.

Ø query and set properties of virtual CDs.

Ø lock and unlock virtual drives.

All functions of this API can be accessed using a COM or C Interface. This lets you use the functions from the most common programming languages. To give you a better idea of how you can get the most out of the API functions, a few examples are included.

The Virtual CD API supports the following operating systems:

- Windows 98
- Windows Me
- Windows NT 4 with SP6
- Windows 2000 with SP2
- Windows XP (Home and Professional)

NOTE: Virtual CD v5 or higher must be installed to use the Virtual CD API.

C Interface

The C Interface files of the Virtual CD API could be found into the <VCD>\API directory. The API functions are used to access the virtual drives and CDs (**vc5api.dll**).

Files for using the C Interface

Folder	Files
<VCD>\API\include	vcdapidefs.h
<VCD>\API\lib	vc5api.lib
<Windows>\system32	vc5api.dll

COM Interface

The COM API Interface of the Virtual CD API enables access to all the required functions for accessing virtual drives and CDs (vc5api.dll). During the Virtual CD installation process, the Interface is registered.

The **vc5api.dll** file for using the COM interface of the Virtual CD API can be found in the **<Windows>\system32** directory.

Examples

This API includes a few examples to demonstrate the use of API functions. These examples can be found in the <VCD>\API\Examples folder.

MFC Examples	Description
VcdApiTest	Direct testing for some of the SDK functions.

The MFC examples can be started directly from <VCD>\API\\Examples\MFC.

VBS Examples	Description
VCDAPIEjectAllDrives	Eject all virtual drives.
VCDAPIGetDrives	Query the virtual drive letters.
VCDAPIGetProperties	Query the properties of a virtual CD.
VCDAPIInsert	Insert a virtual CD.

The VBS examples can be executed directly from <VCD>\API\Examples\VBS.

Property Types of a Virtual CD

A virtual CD can have any of several properties assigned, which can be used to store information with the CD or to link commands with the CD. These properties are divided into two categories: properties assigned as a string and properties assigned as a numeric value.

String Properties

To access the string properties, use VCDApiGetImagePropertiesString (C) or VCDGetImagePropertiesString (COM) to query or VCDApiSetImagePropertiesString (C) or VCDSetImagePropertiesString (COM) to set them.

Property	Description	#
VCD_PROPERTY_DEFAULTDRIVE	Drive where the CD was inserted the first time	1
VCD_PROPERTY_DESCRIPTION	Description string of a virtual CD	2
VCD_PROPERTY_PROGRAM	Program to start when the CD is inserted	3
VCD_PROPERTY_WORKINGDIR	Working directory when the CD is inserted	4
VCD_PROPERTY_COMMENT	Comment text for additional information on a CD	5

Numeric Properties

To access the numeric properties, use VCDApiGetImagePropertiesLong (C) or VCDGetImagePropertiesLong (COM) to query or VCDApiSetImagePropertiesLong (C) or VCDSetImagePropertiesLong (COM) to set them.

Property	Description	#
VCD_PROPERTY_HOTKEY	Hotkey used to insert a virtual CD	6
VCD_PROPERTY_EJECTONEND	Eject a virtual CD on session end automatically	7

NOTE: The properties VCD_PROPERTY_DESCRIPTION and VCD_PROPERTY_COMMENT are stored with the virtual CD. Once set, they are available on all computers accessing this virtual CD. All other properties are set only on the current computer, and must be set on each computer individually.

VCD_PROPERTY_DEFAULTDRIVE

Property value : **1**

Description :

The default drive property defines the drive letter in which the CD was inserted the first time. This can be important if a program installed from the virtual CD expects the CD in the same CD drive every time.

VCD_PROPERTY_DESCRIPTION

Property value : **2**

Description :

The description of a virtual CD is a text containing up to 127 characters . It can be used, for example, to explain the context of the virtual CD.

VCD_PROPERTY_PROGRAM

Property value : **3**

Description :

The start program of a virtual CD is launched directly after the virtual CD is inserted. It can be used to start a specified program every time a virtual CD is used.

If you need the virtual driver letter in which the virtual CD is inserted, the variable **%vcddrive%** can be used. Example: "explorer.exe %vcddrive%" starts the Explorer with the current virtual drive letter as an argument.

NOTE: If the virtual CD is stored on a local medium and was already inserted when computer was started, the specified program does not start!

VCD_PROPERTY_WORKINGDIR

Property value : **4**

Description :

If a working directory is defined, it becomes the current directory directly after the virtual CD is inserted. This can be helpful if a special directory is needed to start a program, or if access to data stored in this directory is required.

VCD_PROPERTY_COMMENT

Property value : **5**

Description :

The comment of a virtual CD is a text with up to 2047 characters. This text can be used, for example, to store installation data (serial number of an application, etc.) or additional information with a virtual CD.

VCD_PROPERTY_HOTKEY

Property value : **6**

Description :

The Hotkey is a key combination assigned to a virtual CD which enables direct insertion of this CD. The returned value stores the modifier in HIWORD and the key code in LOWORD.

VCD_PROPERTY_EJECTONEND

Property value : **7**

Description :

If an image is stored on a network device and is **not** configured to be ejected when the computer is shut down, it is inserted every time a user logs on. This can be a problem because all programs defined in the autostart files start on every logon. This property allows you to set an automatic eject for a virtual CD when a session ends.

0 – Do not eject the virtual CD on session end.

1 – Eject the virtual CD on session end.

API Error Codes

Most of the API functions return an error code which can contain one of the following values.

API Error Code	Description	#
VCD_ERROR_INVALIDINDEX	Internal Index error	-1
VCD_ERROR_NONE	Function was successful	0
VCD_ERROR_NOSUCHFILE	A file could not be found	1
VCD_ERROR_PATHNOTFOUND	A path could not be found	2
VCD_ERROR_TOOMANYOPENEDFILES	Too many open files	3
VCD_ERROR_FILEUNSPECIFIC	Unspecific file	4
VCD_ERROR_ACCESSDENIED	System resource access denied	5
VCD_ERROR_NOVCDIMAGE	File is not a virtual CD image	6
VCD_ERROR_HEADERCRC	Invalid header checksum of a virtual CD	7
VCD_ERROR_INVALIDPASSWORD	Invalid password to access a virtual CD	8
VCD_ERROR_OPENTRACKFILES	Failed to open all track files for a CD	9
VCD_ERROR_IMAGECORRUPT	Corrupt virtual CD	10
VCD_ERROR_DEMOEXPIRED	The virtual CD demo version has expired	11
VCD_ERROR_UNKNOWN	Undefined error	12
VCD_ERROR_RESOURCES	System resource error	13
VCD_ERROR_INVALIDDRIVE	Invalid virtual drive	14
VCD_ERROR_INVALIDUNIT	Invalid unit for a virtual drive	15
VCD_ERROR_DRIVELOCKED	Virtual drive is locked	16
VCD_ERROR_COMMUNICATION	Internal communication error	17
VCD_ERROR_ARGUMENT	Invalid argument	18
VCD_ERROR_INUSE	Virtual CD is currently in use	19
VCD_ERROR_EJECTFAILED	Could not eject a virtual CD	20
VCD_ERROR_NOADMIN	Admin permissions required	21
VCD_ERROR_NOPLAYER	Virtual CD player not started	22
VCD_ERROR_SHARING_VIOLATION	Sharing violation	32
VCD_ERROR_INVALIDPROPERTY	Invalid property type	100
VCD_ERROR_REGISTRY	Registry incorrect	101
VCD_ERROR_REBOOTREQUIRED	Reboot needed to finish an operation	102