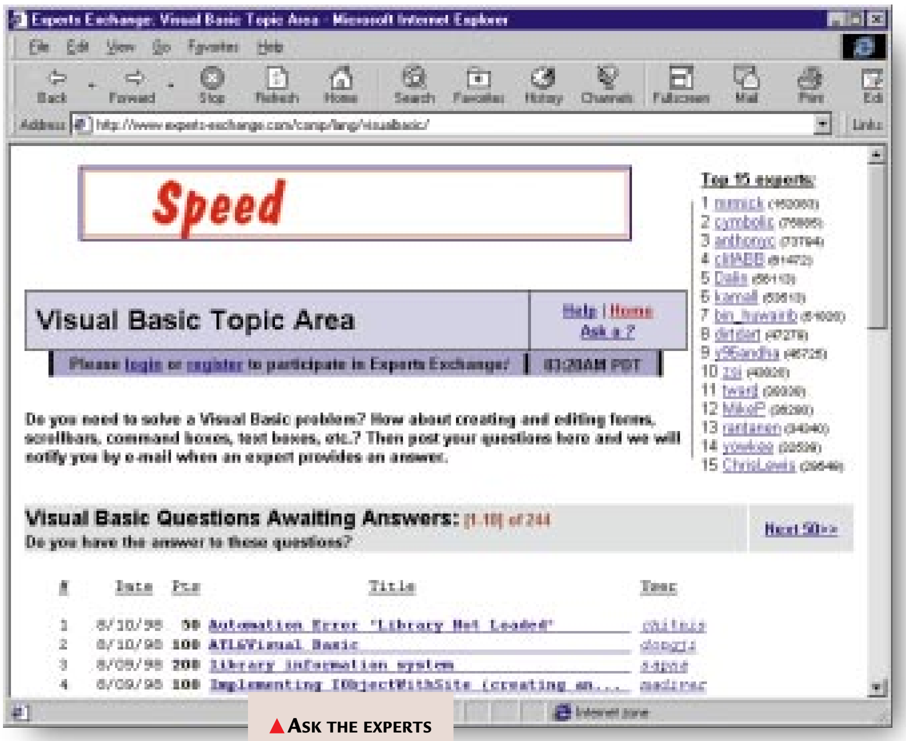
Help is at hand

Tim Anderson investigates the best avenues of help with tricky programming matters, using hints, tips and solutions from readers.

Developing software is a curious mixture of simple and easy visual programming, with the labyrinthine complexity that lies beneath. Few products now come with chunky manuals which disguise the vast size of their programming languages, class libraries and complex development environments. Most of us have to deal with Windows as well: the original API, already notorious for its complexity way back in version 3.0 days, now pales in significance beside the vast arrays of database APIs, COM documentation, and specialist APIs like TAPI, MAPI, ISAPI and a whole lot more. Next, I should mention the internet RFCs, essential for serious web development, along with HTML, Dynamic HTML and XML.

Debate débacle

That means we all need help. Newsgroups are great, but many questions go unanswered and you soon tire of the endless circular, repetitive debates, usually about Microsoft and Java. Incidentally, the vendor-specific groups are usually better, particularly those run by Microsoft and Inprise, if you are after technical help rather than debate. But newsgroups are hardly sufficient. Expensive support contracts, or paid-for support calls, may be part of the answer for corporate sites, but ultimately even these have limitations. The service is not always quick, and the quality of support is variable. Here's a quick look at some other possibilities:



Experts Exchange: Visual Basic Topic Area - Microsoft Internet Explorer

File Edit View Go Favorites Help
Back Forward Stop Refresh Home Search Favorites History Channels Fullscreen Mail Print Edit
Address http://www.experts-exchange.com/comp/lang/visualbasic/
Links
Speed
Internet zone

Visual Basic Topic Area

Help | Home
Ask a ?

Please login or register to participate in Experts Exchange! | 03:20AM PDT

Do you need to solve a Visual Basic problem? How about creating and editing forms, scrollbars, command boxes, text boxes, etc.? Then post your questions here and we will notify you by e-mail when an expert provides an answer.

Top 15 experts:
1 mmmick (160080)
2 cymbolic (75885)
3 anthonyc (73794)
4 clifABB (61472)
5 Dalin (56110)
6 kamall (50610)
7 bin_huwairib (51020)
8 dirtdart (47279)
9 y96andha (46725)
10 zsi (40020)
11 tward (36036)
12 MikeP (36280)
13 rantanen (34040)
14 yowkee (32530)
15 ChrisLewis (29646)

Visual Basic Questions Awaiting Answers: [1-10] of 244
Do you have the answer to these questions?

Next 50>>

#	Date	Pts	Title	User
1	8/10/98	50	Automation Error 'Library Not Loaded'	chitnis
2	8/10/98	100	ATL&Visual Basic	dongzi
3	8/09/98	200	library information system	sapna
4	8/09/98	100	Implementing IObjectWithSite (creating an...	nadirpac

▲ASK THE EXPERTS IS A GREAT IDEA IN PRINCIPLE — UNTIL YOUR MONEY RUNS OUT, THAT IS

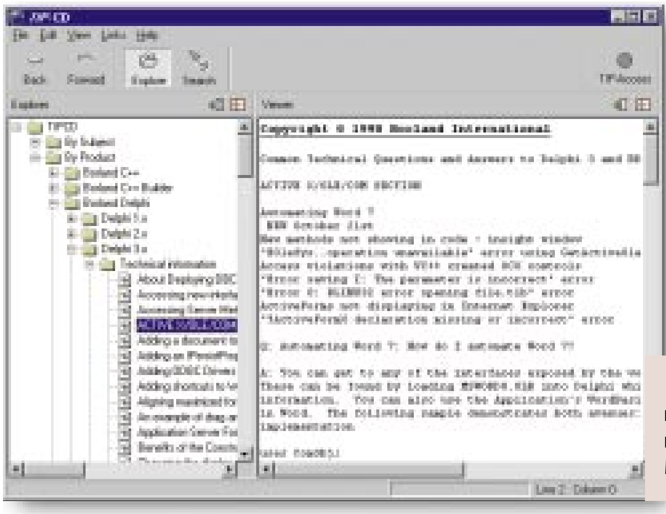
Experts Exchange

Reader Matt Baker writes: "I think you could plug the Expert Exchange in your column, as I find this a useful source of advice. It is at www.experts-exchange.com/."

I took a quick look at this site, which is a regulated version of what often happens on newsgroups but with a commercial twist. You can ask questions and read answers not only on programming topics, but also many others. If you have more answers than questions, then you can sign up as an expert and eventually get paid for your answers. The snag is that although basic access is free, there is a limit to how many answers you are allowed to read, unless you pay. It is an interesting approach if you cannot stand the hurly-burly of newsgroup life, or hanging on the telephone for support.

Borland Tip

Regular readers will know my great enthusiasm for the **MSDN (Microsoft Developer Network) library**, a CD of Microsoft's documentation and knowledge base for Windows, and all its other professional products. And although the subscription is expensive, much of it is freely available on Microsoft's web site if you can get a good connection. The library is also shipped with all the language



Copyright © 1998 Borland International

Common Technical Questions and Answers to Delphi 3 and 98

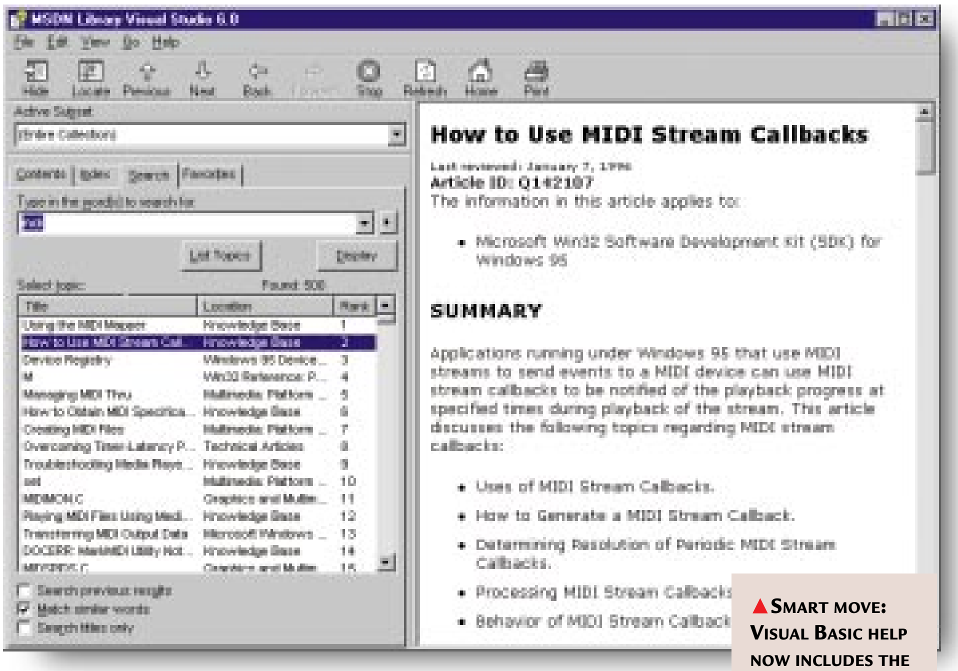
ACTIVE X/OLE/COM SECTION

Automating Word ?

Q: Automating Word ?: How do I automate Word ??

A: You can get to any of the interfaces exposed by the ... These can be found by loading MSWORD8.OLB into Delphi ... information. You can also use the Application's WordBasic ... in Word. The following sample demonstrates both ... implementation.

◀BORLAND TIP LOOKS USEFUL, BUT IT'S NOT A PATCH ON MSDN products including Visual Basic and Visual C++. In the latest Visual Studio, it is completely integrated with online help. For some time I have hoped that Borland/Inprise would come up with something similar for Delphi. The CDs from the Borland Developer Conference are usually made available but conference CDs are rarely useful, except as a memento for delegates, because the presenters do not have time to create proper online versions of their sessions.



Title	Location	Rank
Using the MIDI Mapper	Knowledge Base	1
How to Use MIDI Stream Cal...	Knowledge Base	2
Device Registry	Windows 95 Device...	3
M	Win32 Reference: P...	4
Managing MIDI Thru	Multimedia: Platform ...	5
How to Obtain MIDI Specifica...	Knowledge Base	6
Creating MIDI Files	Multimedia: Platform ...	7
Overcoming Timer-Latency P...	Technical Articles	8
Troubleshooting Media Playe...	Knowledge Base	9
set	Multimedia: Platform ...	10
MIDIMON.C	Graphics and Multim...	11
Playing MIDI Files Using Medi...	Knowledge Base	12
Transforming MIDI Output Data	Microsoft Windows ...	13
DOCERR: MaxMIDI Utility Not...	Knowledge Base	14
MIDIPRNT.C	Graphics and Multim...	15

□ Search previous results
☑ Match similar words
□ Search titles only

How to Use MIDI Stream Callbacks

Last reviewed: January 7, 1996
Article ID: Q142107
The information in this article applies to:

- Microsoft Win32 Software Development Kit (SDK) for Windows 95

SUMMARY

Applications running under Windows 95 that use MIDI streams to send events to a MIDI device can use MIDI stream callbacks to be notified of the playback progress at specified times during playback of the stream. This article discusses the following topics regarding MIDI stream callbacks:

- Uses of MIDI Stream Callbacks.
- How to Generate a MIDI Stream Callback.
- Determining Resolution of Periodic MIDI Stream Callbacks.
- Processing MIDI Stream Callbacks
- Behavior of MIDI Stream Callback

▲ **Smart move: Visual Basic help now includes the whole Developer Network library**

I was interested, therefore, in a new initiative, Borland Tip. Like MSDN, this is on subscription and provides an information CD with the data mostly stored in a Paradox database and viewed through a BDE-based viewer. There are product patches, links to an online Tip site, and email posting of technical news.

If you are a Windows developer, whatever you do, do not get Borland Tip instead of MSDN

Inprise should be encouraged, but I do have some reservations. Like most serious Delphi developers, I had already downloaded all the TI (Technical Information) sheets freely available from the Inprise web site, and much of the first Tip-CD seems to duplicate these. Also, it is not clear to me why Inprise has devised yet another document-viewing application when there are other good (or even better) ones already available — not least the humble Windows help. And, it strikes me that Inprise can only benefit if people know how to use its products, so the relatively high subscription price surprises me. Why not make it a nominal price? If you are a Windows developer, whatever you do, do not get Tip-CD instead of MSDN. To take an example, there is nothing useful on the first Tip-CD about programming MIDI, while MSDN has a number of hands-on articles. Although MSDN has never heard of Delphi, many queries turn out to be Windows queries rather than anything specific to a particular language or product. Even so, Tip has great potential if Inprise is serious about making it good.

THE NAME OF THE GAME

Thanks to David Brown, who has written a **detailed letter about naming conventions**, part of which reads: "I am convinced that the single most important aspect of programming is using well-chosen names. If a program is written with well-chosen names for types, variables, procedures, modules, components, and all other identifiers, it is at least half-way to being a 'good' program. If it is written with names like 'form1', 'edit1', and 'temp', it is guaranteed to be a 'bad' program. By 'good', I am referring to programs which are easy to work with, understand, modify and update.

"It is important to be consistent with names, but conventions used are ultimately a matter of personal programming style. I generally tend to use the Hungarian convention ('edtFirstName', 'lblTelephoneNo', etc) for components; for normal variables, I always start the names with a small letter; procedures and functions I name with an initial capital.

"It is a common misconception that the key to readable code is the good use of comments. In fact, good layout and well-chosen naming systems make comments almost unnecessary. For new programmers, it is difficult to see the need for proper names since most documentation and books use the default 'Edit1' names — this is unforgivable laziness on the part of the authors. In your column, you have the opportunity to reach many beginners and experts, and thus I would urge you strongly to promote the use of naming to help people improve their programming."

David is too polite to say so directly, but I generally use default component names in this column. It is not just laziness, since the default names have the advantage of making it quicker for you to reproduce the example, as well as making it obvious what the controls are. For instance, you will know that Edit1 is an edit box, while Label1 is a label.

But I agree that well-chosen names are essential to good code, although, it has to be said, not everyone likes "Hungarian".

Questions & answers

Q I have been programming in VB for about five years but just recently have started writing a database application. I can select an entry from a ListBox (list of names from a field in the database), pass the selection to an SQL statement, and display the data. I have a problem when the selected text includes a single quote (e.g. O'Reilly). This produces an error. I reckon this is due to the single quote mark in the name, but how can I overcome this?

George McNaney

a *There are a couple of ways around this kind of problem. First, in this scenario it is not clear to me why you would need to include the name in the SQL string. From what you say, I guess you have the list box bound to a database table, and the user clicks on a row to reveal more data for the chosen record. Rather than using the name field to identify the record, it would be better to use an ID field like a number or code.*

Set the properties of the databound listbox as follows:
Rowsource: The data control bound to the address table.
Listfield: The field for display in the list, in this case Name.
BoundColumn: The ID field.
Now your search can be something like this (as one line of code):

```
sSQL = "SELECT * FROM
addresses WHERE
AddressID = " &
DBList1.
BoundText
```

But coming back to your original question, there are certainly occasions when you need to search by a character string which may contain awkward characters such as quotes or other delimiters. Sometimes you can fix it by using the Chr() function to force VB to include the character in the string. For example (one line of code):

```
sSQL = "SELECT * FROM
addresses WHERE Name
= " & Chr(34) &
DBList1.Text &
Chr(34)
```

Chr(34) returns a double quote. Of course, this would still fail, though, if the name itself contained a double quote. It may be unlikely, but the best applications usually allow for such scenarios. The solution is to double-up the problem character.

JET SQL has a feature whereby if you are using a single quote as a delimiter, it treats two single quotes as if they were one literal, single quote. So, if you search for O”Reilly, you will find O'Reilly. The same applies to the double quote.

It all gets messy because VB itself follows similar rules. It won't be obvious, when looking at a line of code, which characters are literal, which are VB delimiters or which are SQL delimiters. To achieve perfection, proceed as follows:

- *Decide which character to use as the SQL delimiter.*
- *When you are assembling SQL strings, use the Chr() version of your delimiter. Chr(39) gets a single quote, and Chr(34) a double.*
- *Check the value you are searching for to see if it contains the delimiter character. If it does, double it up using VB's string functions. If you have VB 6, you can use the new Replace function.*

As an example, the following will find the value 6”1', say, in the height field of a doctor's patients table (again, one line of code):

```
sSQL = "SELECT * FROM
patients WHERE
Height = " & Chr(34)
& "6" & Chr(34) &
Chr(34) &
"1'" & Chr(34)
```

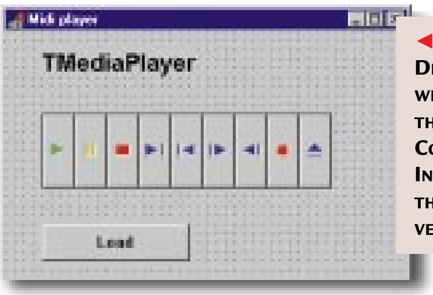
Q Inspired by your Hands On colleague, Steven Helstrip, I've decided to write a small utility in Delphi that lets me alter the parameters of my Yamaha SoftSynth (wavetable software). However, the Delphi help system doesn't include examples of how I should go about sending messages to the MIDI device. In particular, I'd like to alter parameters for Resonance, Reverb, Chorus etc, possibly while a separate piece of software plays a MIDI file. I'm using Delphi 1.

Jonathan White

a *Since Windows 3.1, the Windows API has offered two ways to control multimedia devices. The first is called MCI (Media Control Interface) and is nice and easy. MCI treats WAV files, MIDI files and CD Audio as varieties of multimedia device that can each be opened, played and paused, for instance. All the MCI functions are prefixed mci. For MIDI files, you can think of MCI as an interface to a simple sequencer; easy to use but unsophisticated. Delphi's media player component is an encapsulation of MCI. Devices are identified by a DeviceID, which is why TmediaPlayer has a DeviceID property. The API function mciSendCommand or mciSendString gives you a certain amount of control, for instance, over the tempo.*

When MCI is insufficient, you can use the low-level MIDI Audio services. For example, there is MidiOutOpen to open a MIDI device, MidiOutShortMsg to send a standard MIDI message, and MidiOutLongMsg to send multiple MIDI events and system exclusive messages. For Windows 95 and NT, you should also look up the MIDI stream functions. To use them, you will need a good knowledge of MIDI and a reference that applies the MIDI device you are using and any system exclusive messages it supports. There is no problem with using these functions from Delphi and they are declared in the unit MMSYSTEM.PAS.

There is a problem with the documentation, though, particularly as the Delphi 1 Windows API help does not include the multimedia API. It is included in the Delphi 3 and 4 help files, although in version 4 Inprise seems to have given up preparing Delphi-specific versions, simply shipping the standard Microsoft API help.



◄TMediaPlayer in Delphi is really a wrapper around the MCI (Media Control Interface). This is the Delphi 1.0 version

PCW CONTACTS

Tim Anderson *can be contacted via the PCW editorial office (address, p10) or email* **visual@pcw.co.uk**

Tip subscription *£346.62 (£295 ex VAT) from Inprise 0800 454065 www.inprise.com*

MSDN *information from Microsoft 0345 002000 www.microsoft.com*