



## Crystal clear?

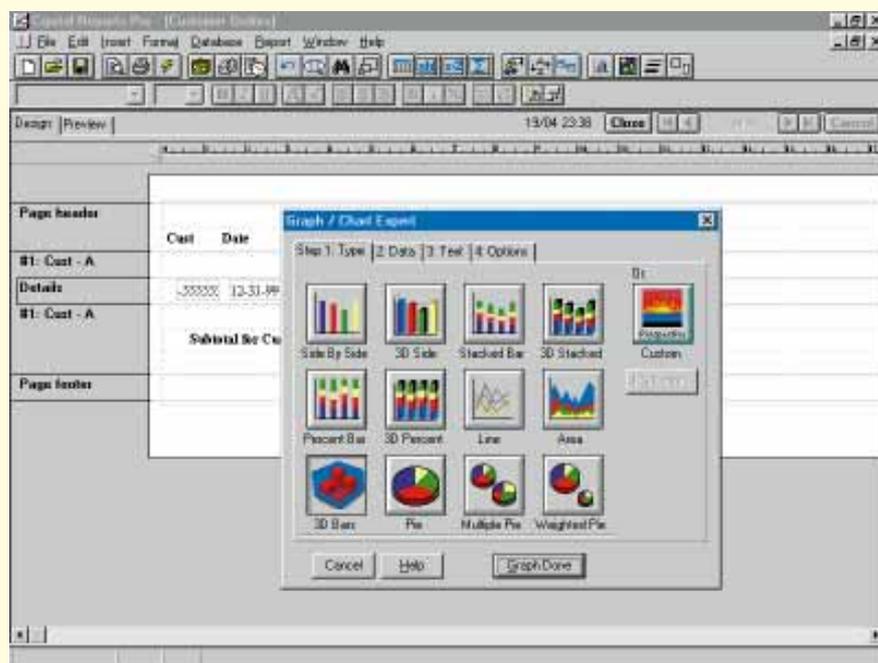
**Tim Anderson** explores the strange world of Crystal Reports and compares the relative merits of Visual Basic 4.0 and Delphi.

Shrouded in corporate mystery is the question of why Microsoft has report writers in both Access and FoxPro but supplies third-party Crystal Reports with Visual Basic. And, why Borland paid good money for ReportSmith, but supplies Crystal with Visual dBase. Somehow Seagate's Crystal Reports has become the most widely used report writer, although not necessarily the most liked. A problem in both Delphi and VB is that the report module is not completely integrated with the main programming environment. For example, Crystal Reports has a formula language which lets you include calculations or text which vary according to values in the current record:

```
if {CUSTOMER.BALANCE} >
{CUSTOMER.CRED_LIMIT} then
"NOTE — credit limit exceeded!!!"
```

Fair enough; formulas are very useful but if you use Visual Basic, you will find that Crystal's formula language is different from VB's. Where VB has VAL, Crystal has ToNumber; Crystal has UpperCase where VB has UCase. There are numerous other examples. All this means more work and, inevitably, more errors. It would be a great improvement if you could call VB functions, including user-defined functions, from Crystal.

Another gripe is that Crystal Reports gives you very little control over its print window, which is important since this is what the user will generally see. The print window contains buttons by default for paging through or printing the report. There are no tooltips on these buttons although it is far from obvious that a suitcase icon means Export. You might want to remove some of these buttons, or design your own toolbar, but you can't. Given the sophistication of VB 4.0's database features in other areas, it is



disappointing. The Visual dBase version of Crystal is better, since it uses the dBase expression builder.

One option is to upgrade to the full Crystal Reports product and I have been looking at version 4.5, the first version for 32-bit Windows. It comes on a CD with two, smart, printed manuals, although they are incomplete and frequently refer you to online help. I tested the release version, although many of the 32-bit database drivers from third parties were not yet available. All the options are covered with VBX, 16-bit and 32-bit OCX, MFC and Delphi VCL wrappers. Those upgrading from the Visual Basic version will find a much improved user interface, with tabbed dialogues and a number of "Experts" to simplify report design.

The Professional edition supplies additional database drivers for SQL

*If you have the full version of Crystal Reports, you can use a variety of graph types in your report*

servers, while the Standard version is intended only for desktop databases like Access, FoxPro and dBase. You can also compile reports to an executable format for distribution. This brings no speed advantage, since it uses the same runtime engine but is useful for giving to those who do not have the full version. There is no gain for developers who will prefer to write their own applications which call Crystal Reports. Another advantage of the Professional version is that reports can be bound to data controls to provide an instant, automatic, reporting facility.

At the heart of Crystal Reports is the CRPE.DLL (or CRPE32.DLL) print

engine. It is noticeable that the OCX and VBX controls are small, because their only function is as wrappers for the print engine itself. This makes it much easier for Crystal to support different programming environments. Unfortunately, Crystal does not expose all the functions of the print engine in the custom controls, so you may find you want to call the DLL functions directly. If you do, you may not use the custom control at all, thus forcing an unpleasant decision on the developer.

Crystal has packed a lot into this product. Things like the visual linking expert, drill-down behind summary reports, the ability to base reports on stored procedures, "top n" reports, integrated graphing and the extensive

formula expression language make this a comprehensive reporting package. It is large, too; the 32-bit print engine DLL is 1,788K and just a little awkward and untidy for developers.

Finally, the full product is not an essential upgrade for VB or Visual dBase programmers, and you should make sure you need the extra features before taking the plunge.

### Delphi or Visual Basic 4.0?

I have received the following email from Bob Osola: "I'm in a quandary about which way to go; I use Paradox for Windows (OPAL) for my data apps, and VB 3.0 Pro for general stuff. The big question is; Do I dump these two and go

## Making sense of groups in Crystal Reports

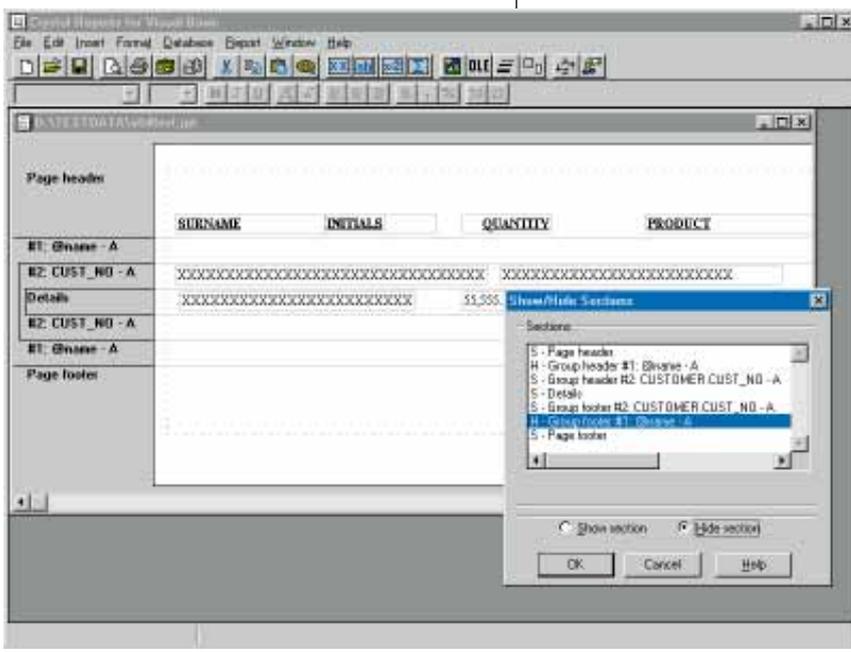
To use groups in Crystal Reports, choose Group Section from the Insert menu and select a field from the active database. Crystal inserts a header and footer area for the group and when the report is run, a new group section will begin whenever that field changes.

Unfortunately, this procedure does not always achieve the desired result. Imagine, for instance, a database in which there is a customer table and an orders table joined by an ID field called CUST\_NO. You want the orders to be grouped by customer and to appear in alphabetical order of surname. If you group on the SURNAME field, all customers with the same surname will be lumped together in one section. But if you group on CUST\_NO, the report will appear in CUST\_NO order, which is not what you wanted.

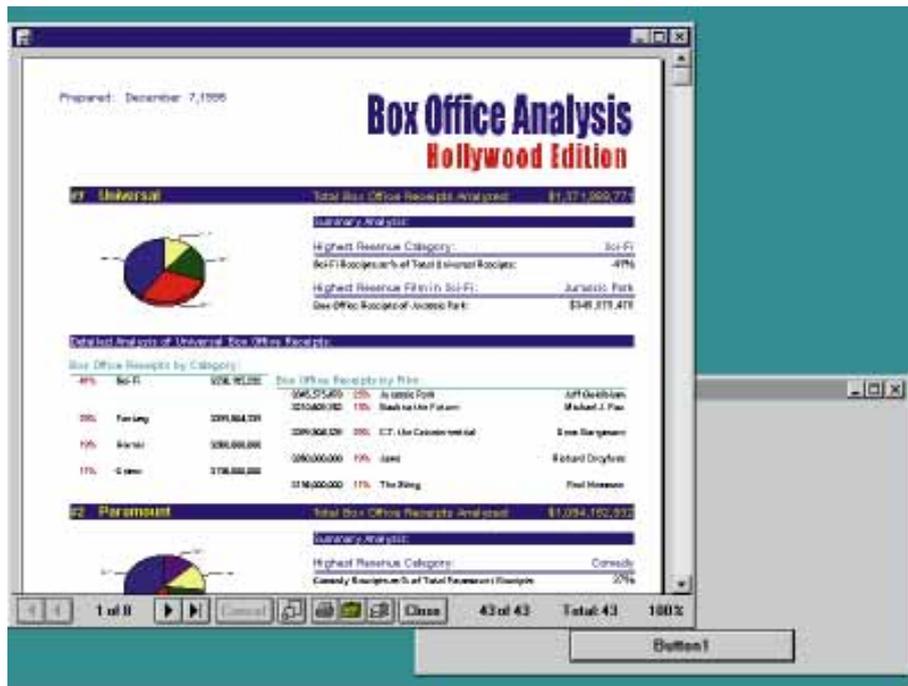
The solution is to create a dummy sub-group. First, insert a group section based on SURNAME. Then insert a further section based on CUST\_NO. Finally, choose Show/Hide sections from the Edit menu and hide the group header and footer for the SURNAME group. Now the report will appear correctly, ordered by surname but grouped by CUST\_NO.

A powerful feature of Crystal is the Group Selection Formula. This lets you filter a report according to the summary values in a group. For example, you might want to see all the invoices above a certain value, where the value of an invoice is only known once all the orders for that invoice have been totalled. You can do this by grouping by invoice, and using the Sum function in the Group Selection Formula.

*Crystal's Hide Sections feature lets you sort a report by one field, while grouping data on another*



## Using Crystal Reports with Delphi



*Using the Crystal VCL component, it is easy to create a Delphi application that displays a report*

The supplied VCL component has not been produced by Crystal, but by a programmer called John Murphy who "needed it for a project I am working on." Documentation is rough and ready, and the author remarks that: "This is an initial build... I will update it as it matures." Better than nothing, but with Delphi's gathering momentum it's a shame that Crystal could not have taken a bit

more trouble.

Installing the component is done in Delphi from the Options — Install Components menu. The location of the component must be added to the library search path and then UCRPE.DCU can be added to the component library. Once Delphi has rebuilt the library, the component appears on the Data Access palette.

A simple report application can be created by simply adding the component to a form, setting the ReportName property to an .rpt file and writing code to set the Action property to 1 at runtime.

The finished Delphi application displayed the report noticeably faster

than its VB equivalent, presumably because the VCL component carries less overhead than the VBX or OCX versions.

Although Delphi comes with its own report component, the Crystal alternative could prove valuable to developers with an existing stock of .rpt files, or anyone wanting to report on .mdb databases, which are handled well by Crystal.

for Delphi, or upgrade to VB 4.0?

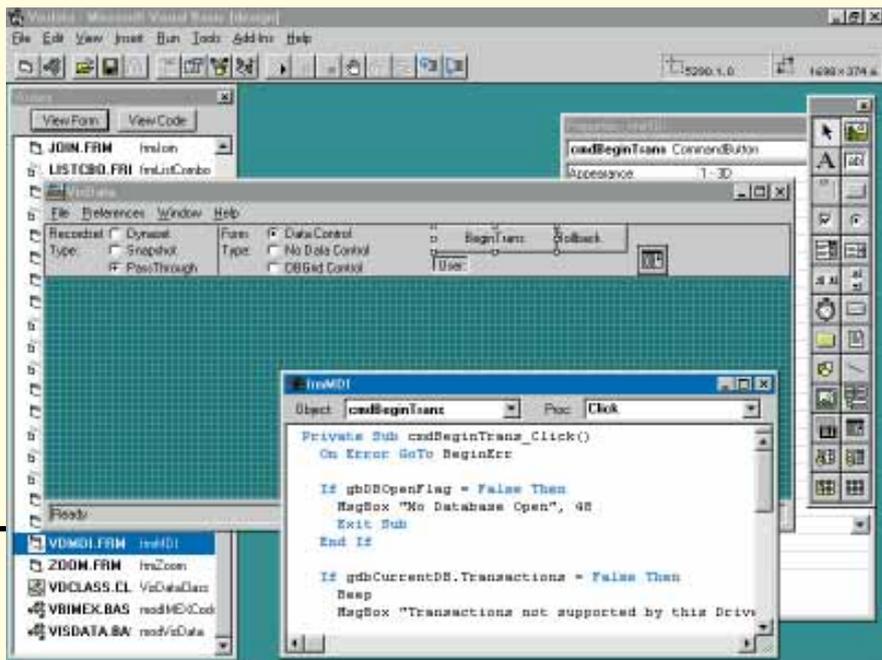
"I'm worried that the VB 4.0 OCX route will both bloat my programs and require massive horsepower (and RAM) to run acceptably — this is presently the main disadvantage with OPAL, which otherwise runs rings around VB and Access for database apps.

"Is there any chance of your running a comparison article on Delphi and VB 4.0? I'm sure many others are harbouring doubts about the VB 4.0 OLE-based future (in my experience; generally flaky and slow) but would sooner stick with the massive user base of VB rather than make a leap of faith to Delphi.

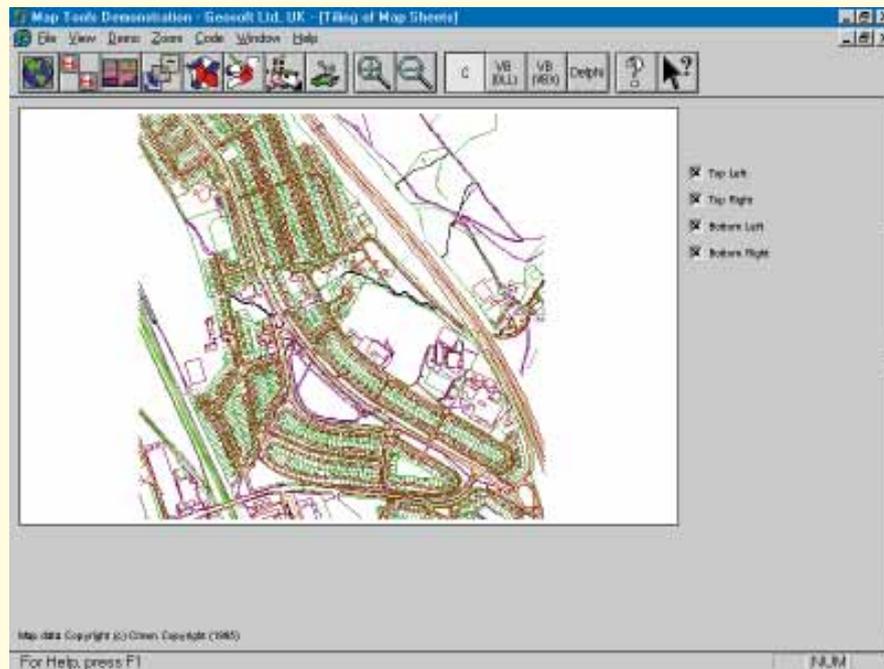
"Having invested a lot of time in learning OPAL, which now appears to be rapidly going out of fashion, I am biding my time to see how Borland runs with Delphi before jumping in. At least VB will still be around in a couple of years time. Any comment or feedback would be much appreciated."

Bob's question is right on target. Despite its massive user base, Visual Basic is far from secure. Its core runtime DLLs, plus JET, plus one or two OCX controls, result in applications that are quick to develop but demanding to run.

Monolithic database applications like Paradox, Access and FoxPro suffer from the same disease. Historically, the solution has been to upgrade the hardware, treating Pentium machines with 16Mb RAM as the new entry-level standard and



*Too big and bloated? Visual Basic 4.0 with its supplied Visual Data application*



leaving older PCs to run old software.

No doubt Microsoft is counting on this process to make software like VB 4.0 and Office 95 acceptable. But Delphi has changed the rules. By combining VB's ease of use with a native-code compiler, it provides acceptable performance on modest hardware, like an 8Mb 486, and blazing speed on fast new machines.

Visual Basic is becoming hard to recommend unless you need some particular feature like the JET database engine, or remote data objects. The same applies to Paradox, despite the elegance of its ObjectPal language. Delphi uses the same database engine and gives better performance and flexibility.

Although Delphi is technically excellent, some people hesitate because it's a relatively new product whose language (Pascal) is less well established than Basic, C or C++, and because its future development depends on Borland (the company that delayed dBase for Windows, dropped Sprint, and sold Quattro Pro).

Nobody can predict the future, but Delphi has become Borland's star product and it is hard to imagine the company letting it go. Delphi 2.0 is 32-bit and brings in support for OCX, OLE automation as client and server, and multithreading, to mention just three key features. The compiler is now shared with Borland C++ and the database engine is used across the range, thus reducing the development effort needed. Delphi is as safe a choice as any development language and the pressure is on Microsoft and others to catch up with its technology.

*GeoSoft's Map Tools lets you add mapping features to VB, Delphi or C++ applications*

#### So what's new?

- Sheridan's VB Assist 4.0 has been released, but its first release is almost as unstable as the same company's first cut of Designer Widgets 2.0. Look out for a review of the patched version soon.
- Microhelp's OLE Tools 5, with 55 OLE controls, and VB Tools 5 is expected shortly, as is Sheridan's ClassAssist which promises to improve Visual Basic's object orientation.
- Graphics specialist Bits Per Second has taken on a new product called Map Tools (illustrated above), developed by Geosoft, which provides mapping functionality via a DLL and VBX control.
- Out too is TopSpeed's Clarion for Windows 32-bit edition. Like Delphi, Clarion features a native code compiler along with integrated database features that make it attractive for some applications.

#### PCW Contacts

**Tim Anderson** welcomes your Visual Programming comments and tips. He can be contacted via PCW at the usual address, or on

**freer@cix.compulink.co.uk**

Contemporary Software **01727 811999** is distributor for Crystal, Sheridan and Microhelp

GeoSoft's Map Tools is available from Bits Per Second **01273 727119**

Clarion is from TopSpeed **01582 763200**