# Using VisioModeler

Terry Halpin

In addition to Microsoft's ORM Source Model Solution in Microsoft Visio for Enterprise Architects, Microsoft has made freely available a former ORM tool known as VisioModeler, as an unsupported product. VisioModeler works under Windows systems up to Windows XP, but seems not to work under Windows Vista. VisioModeler is based on InfoModeler version 3.1. Models built in earlier versions of InfoModeler (e.g. 2.0, 3.0) will work in VisioModeler, but not conversely. Models developed in VisioModeler may be exported to Microsoft's current and future ORM solutions. To obtain the free VisioModeler download, go to http://download.microsoft.com, search by selecting Keyword Search, enter the keyword "VisioModeler" and select your operating system. This should bring up a download page that includes the title "VisioModeler (Unsupported Product Edition)". Clicking on this will take you to the link for the download file MSVM31.exe. Click on this to do the download. The 25 MB file takes just over 2 hours to download on a 28.8 k modem.

Launching VisioModeler

Once you have installed VisioModeler, you can make a short-cut to it by going to its location, right-clicking on its icon, and choosing Create Shortcut. Drag the short-cut icon to the desktop so you can access it from there in future. If desired, rename the shortcut by clicking on the text box below the icon and entering a new name (e.g. "VisioMdlr"). Double-click this icon to launch VisioModeler. Alternatively, from Start on the Task Bar hold the left mouse button down and navigate through Programs to VisioModeler and release the mouse button.

The main window

When you start the program, the main window appears. It contains five areas, ordered from top to bottom: title bar; main menu; main tool bar; document window; status bar.
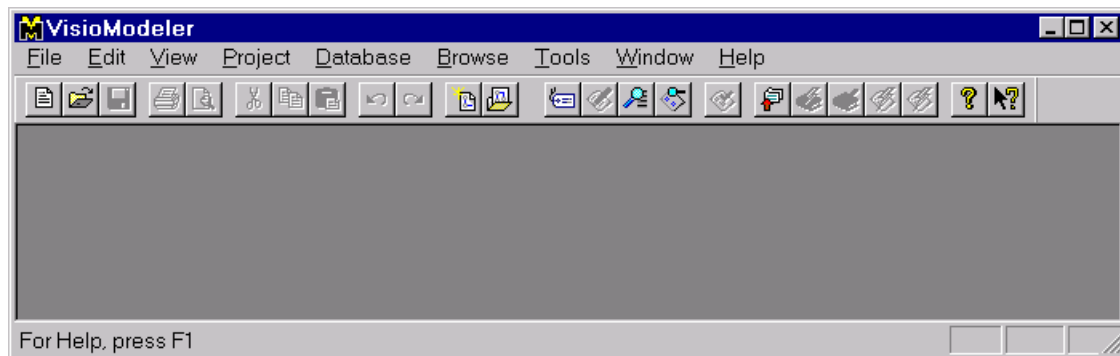
The *title bar* displays the title of the current program, and includes three standard windows buttons, as shown. Following standard windows behavior, a window that occupies only part of the full screen can be moved by dragging the title bar, and resized to any desired size with the resize arrows (⇔, ⇕ …) that appear as you move the mouse over edges of the window.

▭	*minimize window*	Click this and the window becomes a button on the status bar. Clicking that button restores the window.
▭	*resize window*	Click this and the window resizes to fit only part of the screen. The button is redisplayed as ▭. Click that to resize to full screen.
✕	*close window*	Click this to close the window.

The *main menu* lists nine options. Clicking on any of these displays a drop-down menu with further options. Options have one letter underlined. If you hold the Alt key down as you type that letter, this also selects that option. For example, Alt-H invokes the Help menu.

The *main tool bar* includes twenty four buttons. If a button is grayed out, it is currently unavailable. For example, if you open the main window with no document, the Save button ⊟ is grayed out since there is currently no document that you could save.

The *document window* is your work area for the current document. When starting VisioModeler for the first time, this window is blank.

The *status bar* displays various messages about the current state. It is one of five main help mechanisms.

Help

In addition to the printed and online manuals, the following kinds of online help are provided.

- *Tool tips*
  Let the cursor hover over an icon to view a brief description of its function. At the same time, a longer description appears in the status bar at the bottom of the screen.

- *Status bar help messages*
  The status bar displays messages about a menu item or current validation error/warning. It also indicates which, if any, of the NumLock, CapsLock and ScrollLock keys are toggled on

- *Context help button* (↖?)
  Click this (on main menu bar) and the mouse cursor becomes that shape, prompting you to point to the item on screen for which you need help. Click any item, and help about that item appears.

- *F1 key* or *dialog box help button* (?)
  Hit the F1 key or click the help button to view help about your current dialog box or window.

- Help *menu button*
  The main menu Help option allows you to access explanations of almost every aspect of the tool. For example, click on the Help option, then click Help topics, then double-click the Procedures book (or highlight the Procedures book and click Open). Then double-click on the topic that interests you. To exit help, click the close button (⊠) at the top right hand of the help window.

Select a driver

If you haven't already done so, go to the File menu, select Driver Setup and pick an appropriate database driver for the DBMS you wish to use (e.g. Microsoft Access). Although you don't really need to do this before working on a model, it's handy to do it in advance if you want to get down to the detail of physical data types etc. You can change your driver at any time later if you wish. If you are using a feature-restricted version of the software at home, you may have fewer driver choices. In this case, choose a driver with good support for the SQL standard or for the DBMS you want to use.
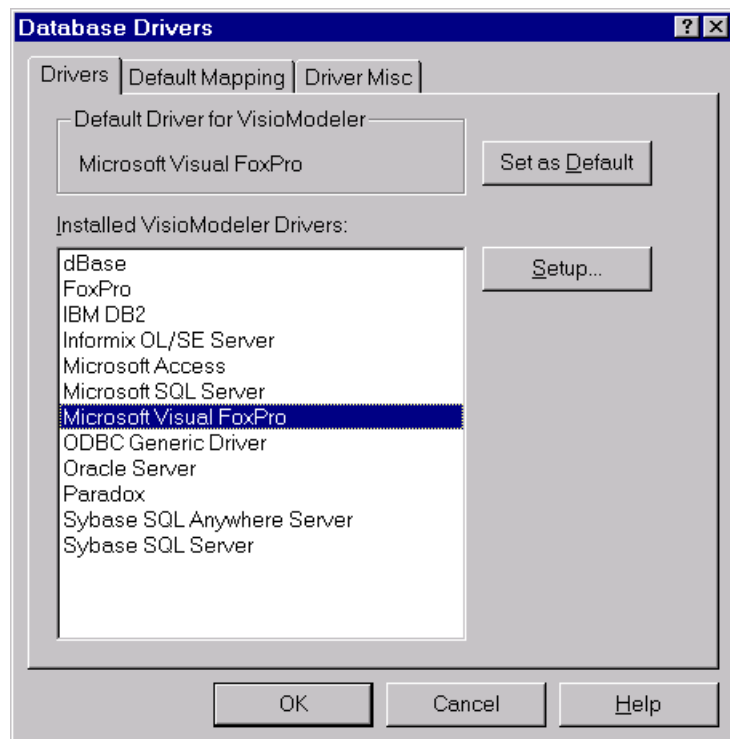
Open an existing document

To open an existing document, such as a model you have worked on or a sample model provided, click the second icon 🗁 in the main toolbar, or click the Open option from the File menu. The Open-document window has a Look-in field to display the current folder (e.g. the Programs folder or the Models folder). It's a good idea to create a Models folder of your own to store your own models.
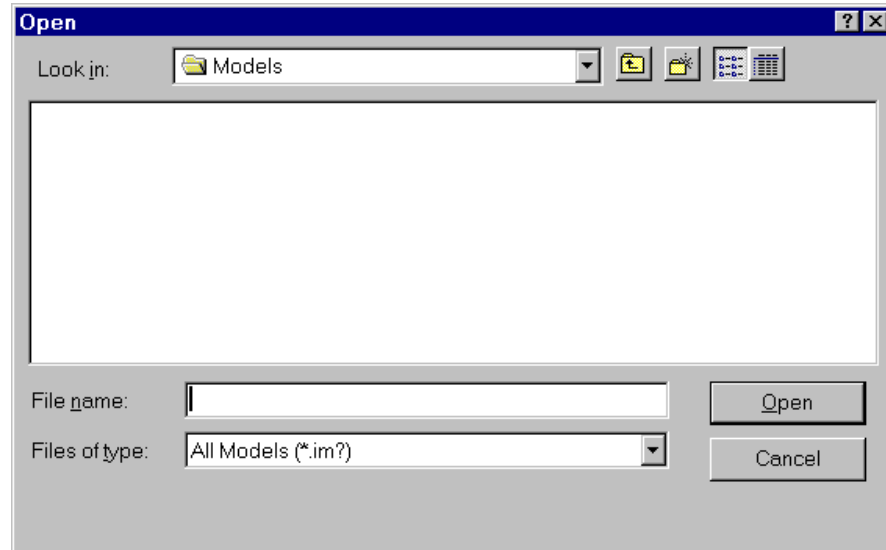
If necessary, change the folder (e.g. double-click the Look-in box, and click till you get it) until your document appears. Then either double-click the document, or highlight it and click the Open button). The sample models are contained in the Samples folder (e.g. go up one level from the Program or Models folder, then down to the Samples folder). For example, to work with the InfoU model, obtain the following screen and then click the Open button.
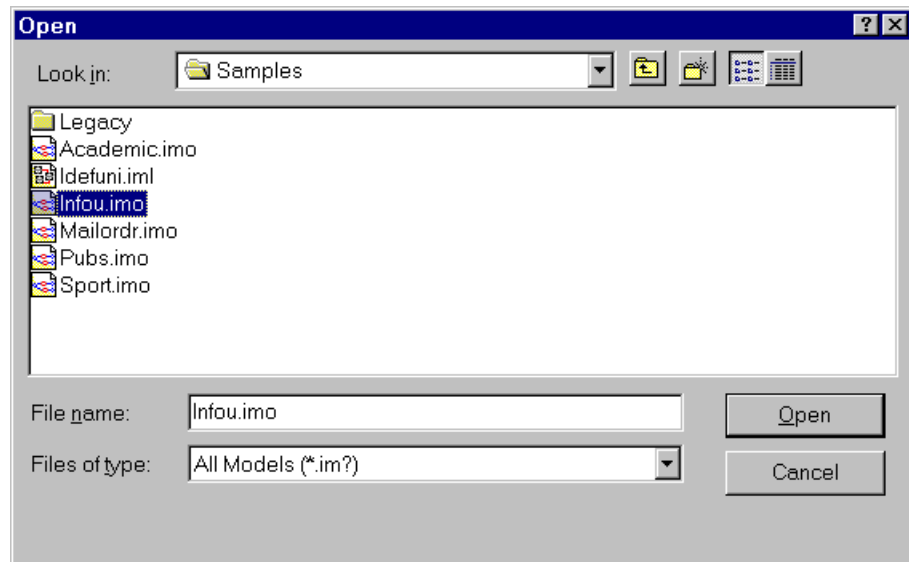
The InfoU model is a sample model about a university, and is spread over four pages. The status bar indicates which page you are on (e.g. "Page 1/4" indicates page 1 of 4), and if the page has been named, its name is also displayed (e.g. "Instructors and Departments"). Open it for practice. Notice the *scroll bar* at the bottom. You can scroll across a single page using the usual arrow buttons. Notice the extra buttons on the scroll bar. You can *move between pages* by using these extra buttons or entering their equivalent keystrokes.

| Action	Button	Keys
next page	Σ	PageUp
previous page	™	PageDn
first page	|™	Ctrl+PageUp
last page	Σ|	Ctrl+PgDn
go to page …	…	Ctrl+G

If you choose the go to page button (**…**), a dialog box opens to select the page you want to go to.

Browsing

You can also navigate between pages by *jumping to other copies of the same object type*. An object type that is duplicated in the global schema is displayed with a delta "▲" symbol. To jump to its other copies, right-click on the object type and an ObjectType-window appears. To find all the pages on which it occurs, click Find References ... and then select the page you want. The desired page now appears with a red arrow pointing to the object type. You can also move to the next copy of the object type by selecting Next Instance, or to the previous copy by selecting Previous Instance. For practice, try this with the Room object type in the InfoU model.

If you want to find where an object types occurs in a model without having to first locate one instance of it, pull down the Browse menu and click Browser ... (Ctrl+B). This opens the *Browser*. In a large project, the browser can be used to browse several component models, but we confine our attention here to browsing a single model, using InfoU as our example.

Hit the Search>> button (below the Refresh button) then enter the name of the desired object type in the Search for field, or scroll down the object pick list and highlight the object type you want. See the next illustration, where we have chosen to look for the Room object type. The browser's verbalization section gives you details about the object type.
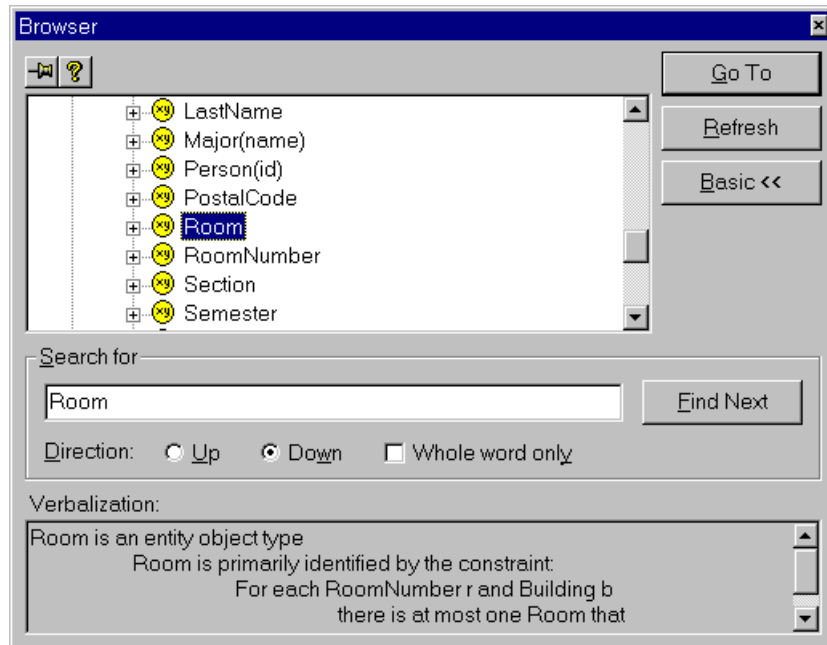
To go to instances of the object type, you can hit the Go To button. If the object type occurs more than once, a Jump-Resolution dialog box opens to resolve which instance we want to see. For example, the following display appears for our Room example. Select the instance you want, and it is displayed with a red arrow pointing to it.
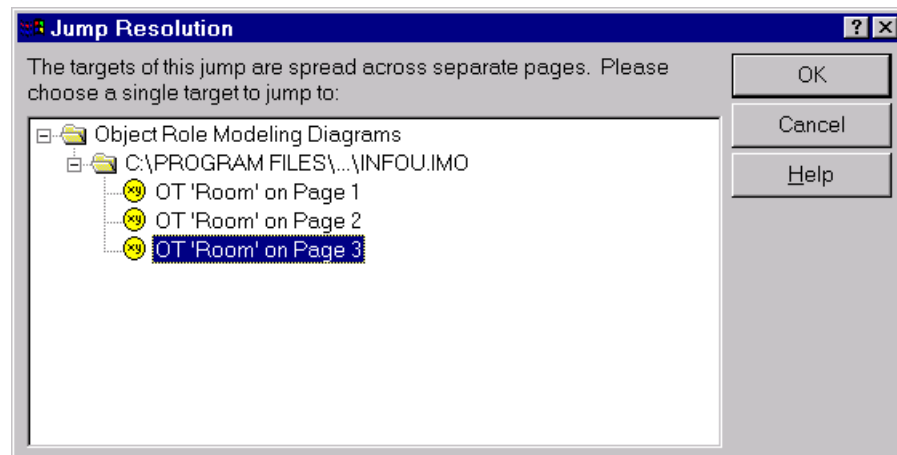
Verbalizer

VisioModeler can automatically verbalize aspects of an ORM or Logical diagram in English. Try this out for yourself. Click the verbalizer icon from the main toolbar, or select Window from the main menu then Verbalizer. You can have any part of your schema verbalized by selecting it (e.g. any constraint, any predicate, any object type). You can select a rectangular window over a subschema by holding the left mouse button down and dragging: all the schema contained in the rectangle is now verbalized.

If the verbalization text in does not all appear in the verbalizer window, you can resize the window and scroll about to read more. The verbalizer window remains open until you close it.

New model

To start work on a new model, click the left-most icon 📄 in the main toolbar, or click the New Model option from the File menu. A wizard presents you with the choice of working at the physical level (Dictionary Model Diagram), logical level (Logical Model Diagram) or conceptual level (Object Role Modeling Diagram). Choose the last, (then Next if it's not grayed out) then Finish. An ORM document window opens. To maximize this, click the resize button □ .

Fact editor

You can textually enter relationship types whose object types are all simply identified, as well as simple uniqueness and mandatory role constraints on them, using the Fact Editor. To invoke the Fact Editor hit function key F5 (or click the FactEditor icon from the main toolbar, or select "Create a new fact" from the Tools menu). You can enter relationship types in freeform or guided mode. Freeform is faster. Start object types with capital letters and enclose reference modes in parenthesis, e.g.

Employee (empnr) works for Department(code)

If you want to enter the inverse predicate at the same time, use a slash "/" to separate the forward reading from the inverse reading, e.g.

Employee (empnr) works for/ employs Department(code)

Here is how that looks in the Fact Editor. The various components of the sentence (object type, reference mode, predicate, slash) are colored differently to help distinguish them. Value types may be entered by appending empty parentheses (signifying no reference mode), e.g. Employee has Empname()
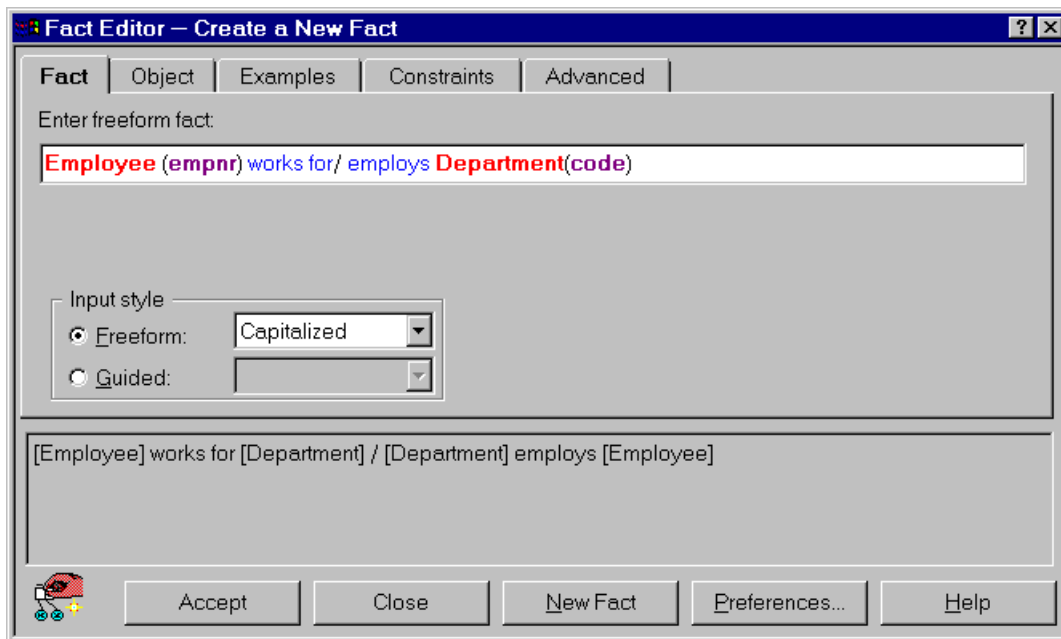
You can enter some examples and constraints by choosing the relevant tabs. Currently the fact editor allows you to enter constraints that apply to just one predicate (e.g. internal uniqueness, simple mandatory, internal frequency and ring constraints). By default, the constraints page combines uniqueness and mandatory to make it faster to specify them. For instance, choosing "exactly one" in the following example means both "at least one" (mandatory) and "at most one" (unique). The constraint symbols and verbalization automatically appear to help you see the result of your choice. If you don't want to use this default shortcut, click the Preferences button and untick the option that indicates combined uniqueness and mandatory (UM).
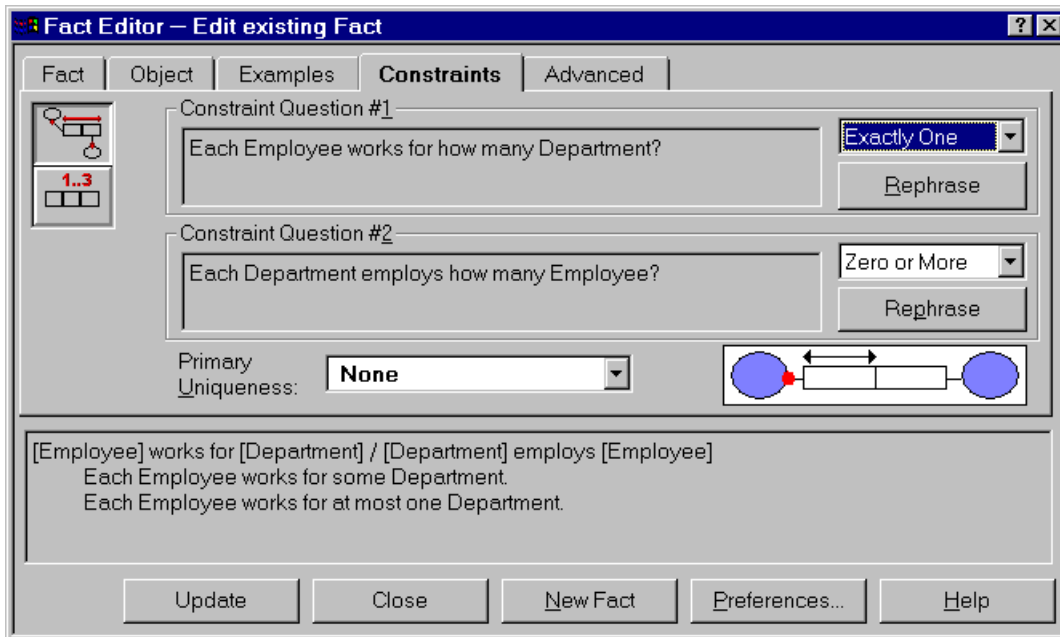
Once you are happy with your entry, click Accept (or Update). In the above illustration, the left-most button is called "Update" instead of "Accept" because I added the constraint some time after I originally entered the fact type. To update a predicate, just select the predicate and hit F5 to open the fact editor on it.

Hitting the Accept/Update key for a new/old relationship type causes it to be automatically drawn/redrawn. You can continue to enter more fact types in this way or close the fact editor window. Once you have supplied the reference scheme for an object type there is no need to repeat it in later relationships.

The fact editor cannot be used to define co-referenced object types, though it can be used for nesting. Other constraints (external uniqueness, disjunctive mandatory, subset etc.) and co-referenced object types should be entered with the diagrammer.

It is a good idea to include examples for all relationship types. Click on the Examples tab and enter enough examples to illustrate the relevant constraints. For example, the following screen shows three fact instances for our Employee works for Department fact type.

Fact Editor — Edit existing Fact

| Fact | Object | **Examples** | Constraints | Advanced |

	Employee	Department
1	e1	CS
2	e2	CS
3	e3	MA
4		

Analyze...   Set UC Constraints

Employee e3 works for Department MA.

Update   Close   New Fact   Preferences...   Help
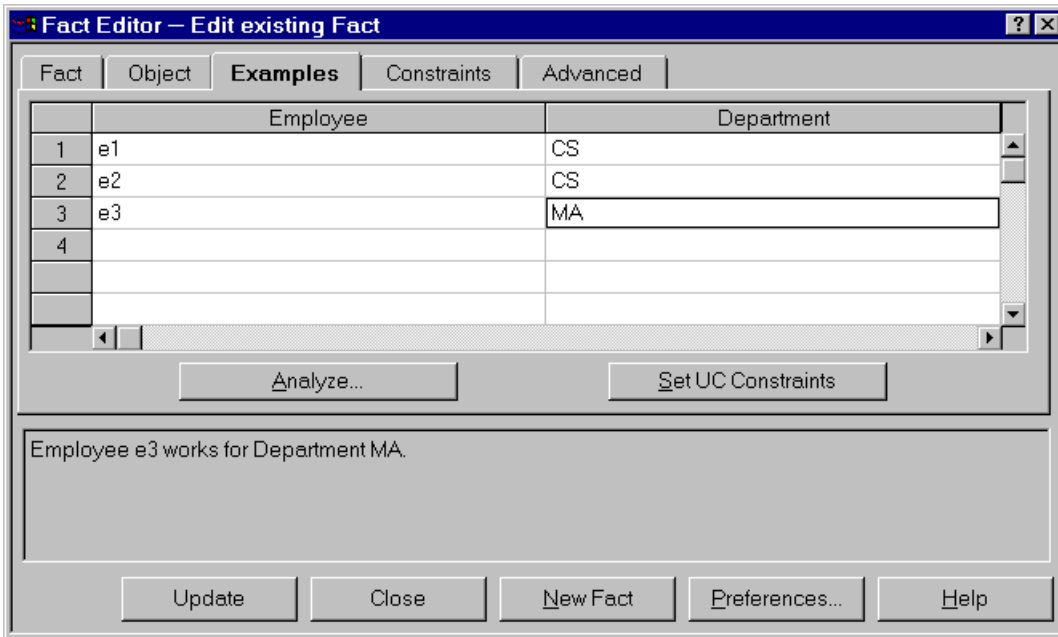
Notice that employees are unique but that departments are repeated. This is consistent with our earlier decision that employees work for at most one department but a department may have many employees.

You can use the Analyze button to request the tool to induce the constraints from your examples, or to check for inconsistencies between your data and your constraint specification. Try it out for yourself. This is a very useful feature for validating constraints.

When you are viewing the constraints pane, only relevant constraint options appear. For example, unless two roles are played by the same object type, you will not see the ring constraint option. Let's see how to add ring constraints by working with the following ring fact type:

Employee supervises/ is supervised by Employee

Enter this fact type then go to the constraints pane. Since we have entered a ring fact type, the constraint buttons now include a ring constraint option (the bottom constraint button). Click this button. The screen should now look as follows.

Fact Editor — Create a New Fact

| Fact | Object | Examples | **Constraints** | Advanced |

Select a Ring constraint to edit or press New to create a new one.

New ...   Edit...   Delete

[Employee] supervises [Employee] / [Employee] is supervised by [Employee]

Accept   Close   New Fact   Preferences...   Help
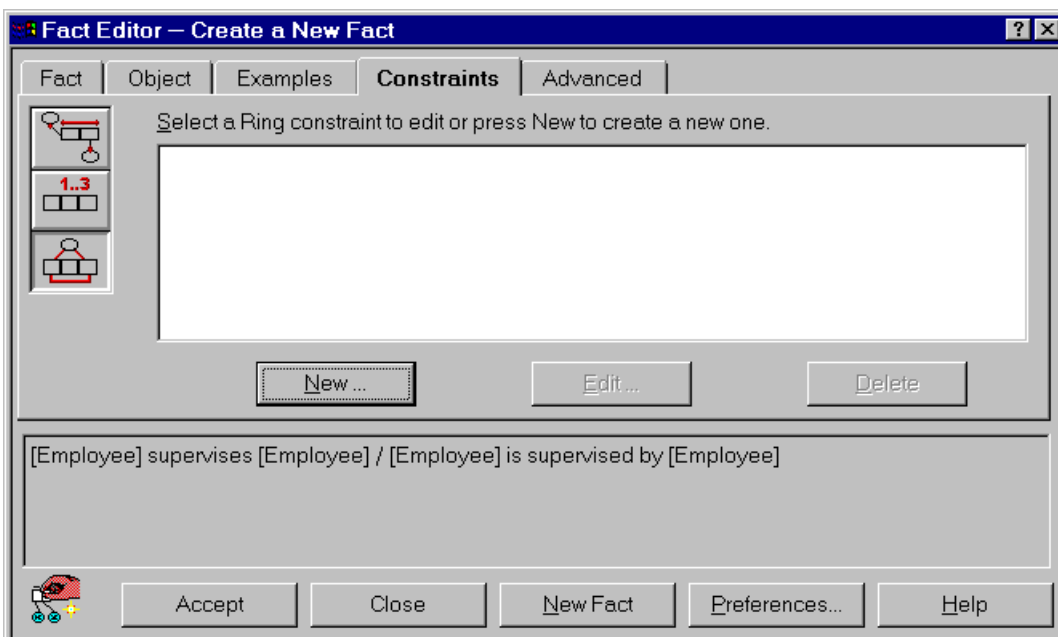
Since this is a new constraint, hit the New button. If a ring constraint already existed, the Edit and Delete buttons would be available for those tasks instead of being grayed out.

The following screen now appears. Although this looks a bit complex, it should actually make your life easier. I designed this interface so you wouldn't have to worry about the redundancies and inconsistencies that might arise from picking wrong combinations of ring constraints (e.g. asymmetry implies irreflexivity, so it would be silly to specify both).

New Ring constraint

Define Ring constraint for 'Employee'

Ring constraint Value: Undefined

Undefined

ans   as   ir   sym
ac   it

Verbalization:

Select a valid ring constaint value to define a complete ring constraint.

OK   Cancel   Help
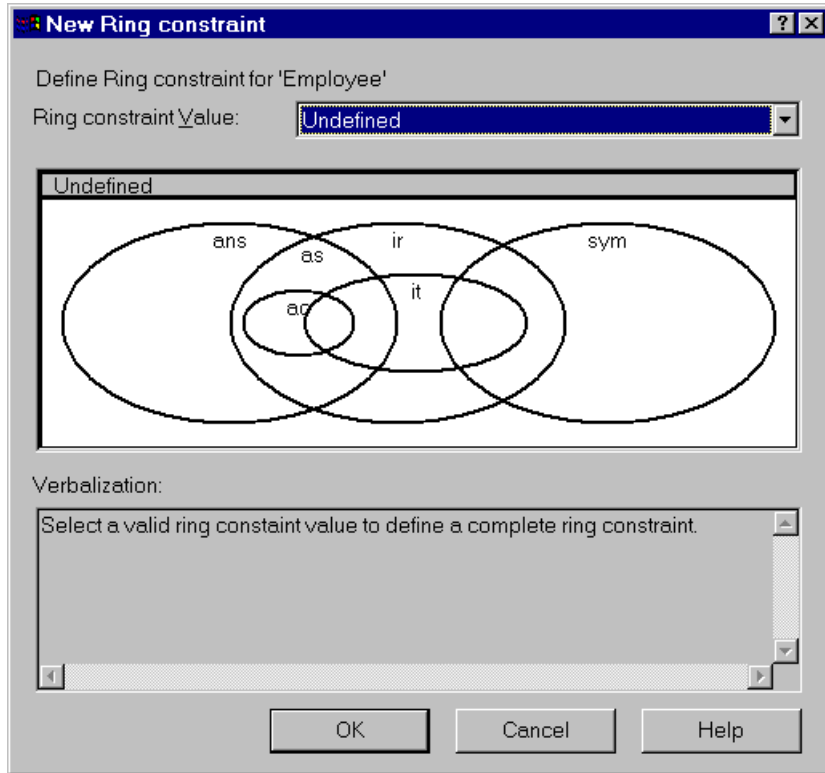
Just pick the constraint option that applies to your case. Let's suppose we want asymmetry. Moving the cross-hair to that option causes the following display. Notice that the word "Asymmetric" appears as the ring constraint value, and the verbalization explains what the constraint means.

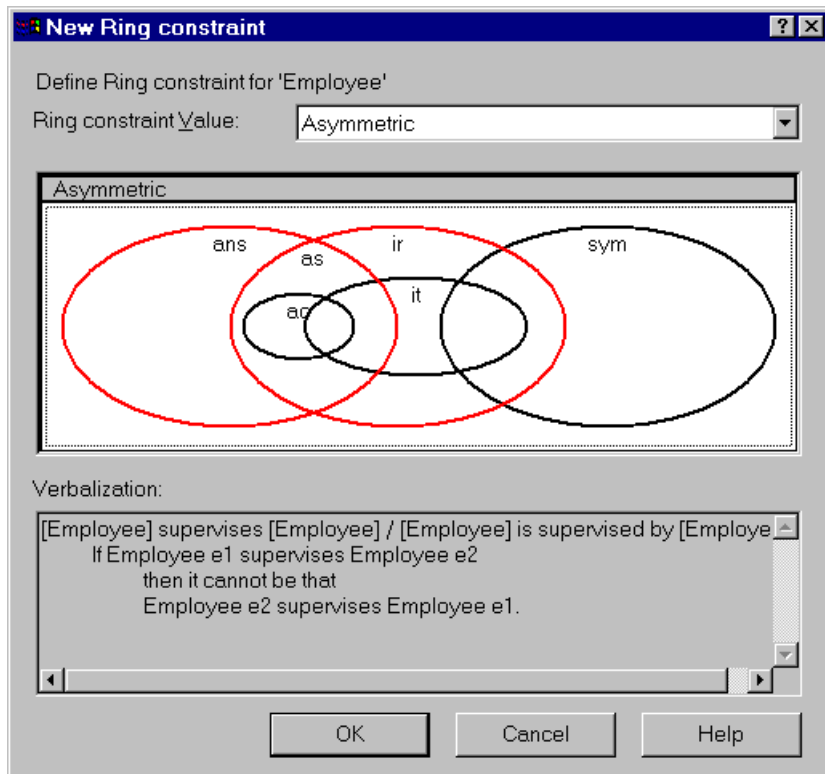Now hit the OK button, then Accept in the Fact Editor, and the ring fact type is drawn for you, including the ring constraint.

New Ring constraint

Define Ring constraint for 'Employee'

Ring constraint Value: Asymmetric

Asymmetric

ans   as   ir   sym
ac   it

Verbalization:

[Employee] supervises [Employee] / [Employee] is supervised by [Employee
        If Employee e1 supervises Employee e2
            then it cannot be that
            Employee e2 supervises Employee e1.

OK   Cancel   Help

Main toolbar

VisioModeler includes five toolbars: main; ORM; constraint; logical; alignment. The icons on the main tool bar are collected into functional groups as follows. Most of these are briefly discussed in this document. Some of these (e.g. New project and Open project) can be ignored for now.

| New/ Open/ Save model | Print/ Print preview | Cut/ Copy/ Paste | Undo/ Redo | New/ Open project | Properties/ Fact editor/ Verbalizer/ Browser | Check | Extract/ Build/ Edit/ Generate/ Alter | Help/ What's this? |
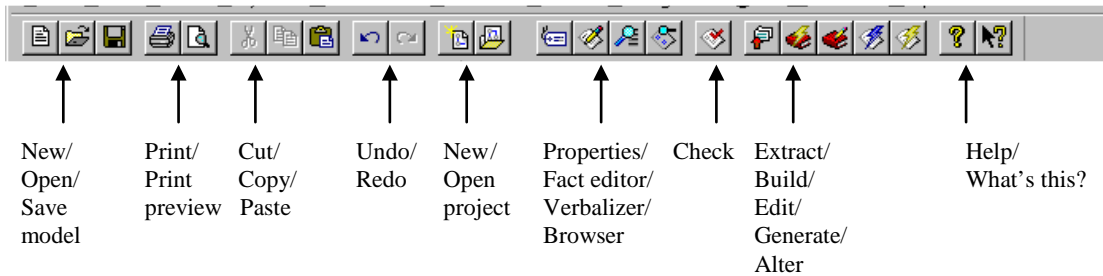
Undo and Redo

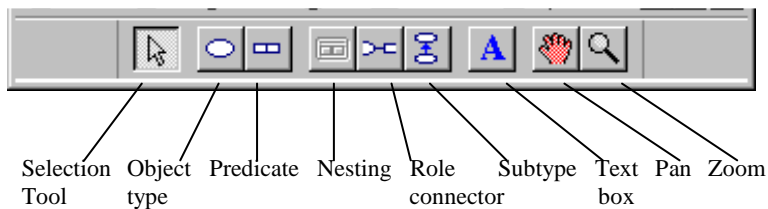You can use the Undo command to reverse your last action, by selecting the Undo button ↤ on the main tool bar or by choosing Undo from the Edit menu or by hitting Ctrl+Z (hold the Ctrl key down as you hit the "Z" key). You can undo any number of actions up till your last save. To repeat an action, select the redo button ↦. The *Escape key* can also be used for undoing the current action.

Diagrammer

A very brief overview of the diagrammer is now given. For more details, see Help or the VisioModeler manuals (especially *Guide to VisioModeler*). Only one copy of the manuals is available for our site, so try the online help first.

Drop down the View menu to see or edit the display of current tool bars, rulers etc. Alternatively, right-click on any tool bar, and a sheet appears with the list. By default, an *object type selector* is turned on. You can drag current object types from this onto the diagram. The selector can be toggled on/off with Ctrl+H.

You can enter any relationship type and any ORM constraint graphically using the Diagrammer, by using the appropriate icon from the ORM tool bar or the Constraint tool bar. Hold the mouse over one of the icons for a second and a tool-tip appears to describe its function, and a longer explanation appears in the bottom status bar. The nine icons in the *ORM Tools tool bar* are indicated below.

| Selection Tool | Object type | Predicate | Nesting | Role connector | Subtype | Text box | Pan | Zoom |

The *selection* tool is used as an ordinary mouse pointer to select the diagram component. To *delete* a component, first select it then hit the Delete key. To add an object type to the diagram, click the *object type* icon, move to the relevant point, and enter the details. You can always add details to an object type or predicate by right-clicking it to open its property sheet. The following illustration shows the property sheet for the Employee object type.

The *Definition* pane is used to enter or edit the object type's name. If an object type is *independent* (formerly called "lazy"), tick the Independent box and an exclamation mark appears appended to the object type's name. This means the disjunction of its fact roles is not implied to be mandatory. You should do this for any object type that is independent (even nested object types). For further discussion, see section 6.2 of my schema design book. You can make use of an object type defined elsewhere by declaring it to be external. In large projects, different object types and predicates with the same name can be distinguished by adding a name space to indicate the local context.

The *RefMode* pane is used to enter the reference mode, and the *Data Type* pane the data type. The *Composite Type* pane is used to specify certain mapping options (e.g. row types) for ORDB (object-relational database) systems such as Informix-Universal Server. Use the *Subtype* pane to enter the subtype definition; tick Map to separate table if you want to override the default subtype mapping (which absorbs functional roles into the supertype table). If the subtype has more than one supertype, use the primary supertype field to declare its primary supertype (for absorption etc.). Ignore the Create table inheritance option unless you are mapping to an ORDB.

There are more tabs than can fit across the screen, so scroll keys (◄, ►) are used to access the other options (this interface will probably be replaced by stacked tabs in a later version). Use the *Value* pane to enter value constraints. Value ranges can be entered by specifying their start value in the From field and their end value in the To field. A list of values may be entered one at a time in the Value field, and clicking the Add button. Entered values and value ranges appear in the Defined Values/Ranges field. You may enter multiple value ranges/lists if in each case you first highlight <NewValue/Range>. In the example, I've added the value constraint for an SQL identifier character (underscore or letter). Entries are removed by highlighting them in the Defined Values/Ranges field and hitting the Delete button.

Scroll further right to find other tabs. The *NestedRoles* pane is relevant only if the object type is nested. The *Notes* tab can be used to document further comments about the object type. The *Graphics* pane is used to position the value constraint and to restrict the number of values in the value constraint that are displayed in the diagram. The Object Type Properties sheet for an existing object type can be opened at any time by double-clicking the object type.

Returning to the ORM toolbar, click the *predicate* icon to add a predicate to the diagram. This opens the Predicate Properties sheet. This sheet can be opened for an existing predicate by double-clicking the predicate. Use the Definition pane to specify the arity (number of roles) of the predicate (2 by default). The illustration shows the pull-down menu for arity.
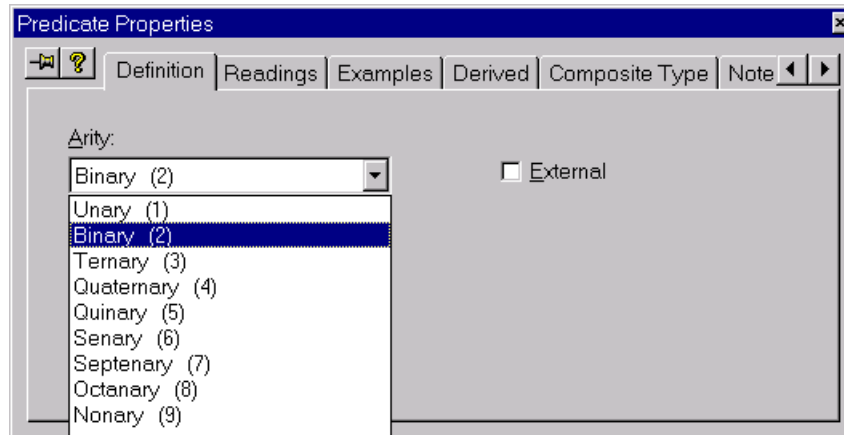
The Readings pane is used to specify various readings for the predicate. An *n*-ary predicate can be given up to *n*-readings (one for each starting role). The "role name" fields may be used to specify names for the relational columns to which the role maps (if you omit these, the tool will generate names for you at mapping time, which you can always edit later).
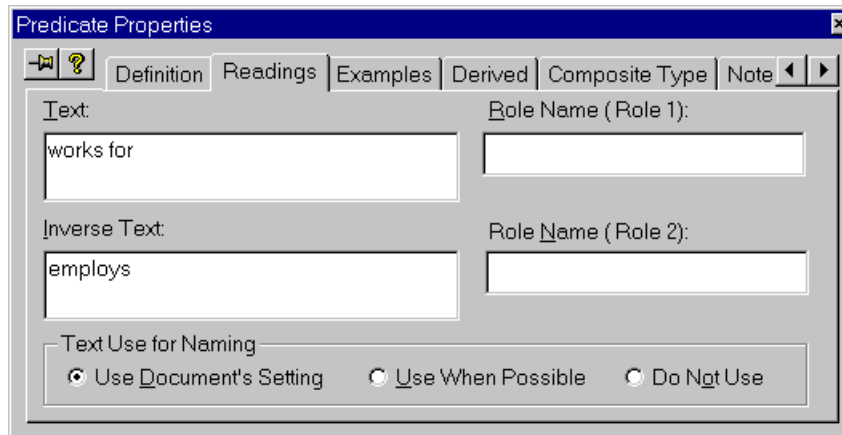
The Examples pane is used to add examples of relationship instances. The Derived pane is used to indicate whether the predicate is un-derived (none), derived only (derived) or derived and stored. The tool displays zero, one or two asterisks on the predicate for these cases respectively. The other panes are easy to master (try it yourself).

Predicate Properties

Definition | Readings | Examples | Derived | Composite Type | Note

Derivation Rule

Rule   Placement: Lower Right

○ None

○ Derived

○ Derived and Stored
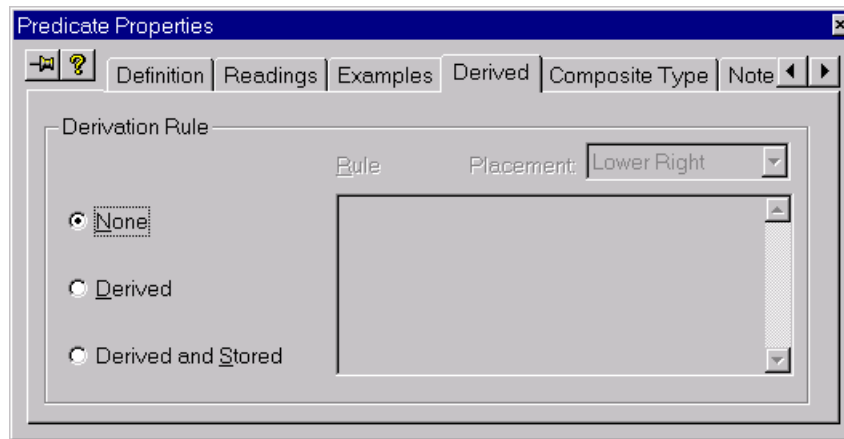
To nest an object type, first select it and then click the *nesting* icon. To connect a role to an object type, click the *role connector* icon, then hold the shift key down as you first click the role, then the object type and then let go. Until you get used to this, the escape key is handy for canceling your current action.

To add a *subtype connection*, click the subtype icon and drag it from the subtype to the supertype. To add extra comments to 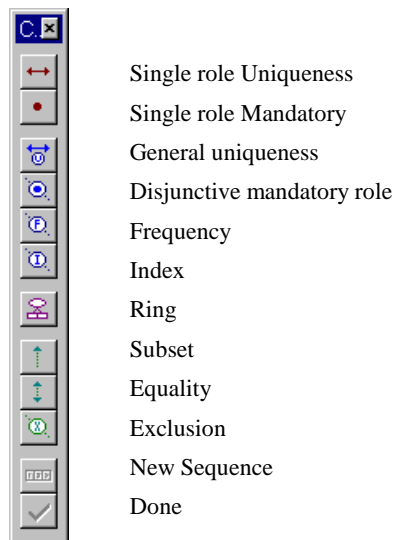a diagram, click the text box icon, move it to the relevant position and enter the text. The *grabber panner* tool can be used to pan around the diagram. The *zoom in/out* tool lets you change the magnification in which the diagram is displayed (you can also access zoom from the View menu).

The constraints palette is used to add constraints to existing ORM predicates. Its main functions are itemized below.

Icon	Function
	Single role Uniqueness
	Single role Mandatory
	General uniqueness
	Disjunctive mandatory role
	Frequency
	Index
	Ring
	Subset
	Equality
	Exclusion
	New Sequence
	Done

To add a *uniqueness constraint to a single role*, click the single role UC icon [Γ] then click the role. If the predicate has more than one uniqueness constraint you may declare one of these as primary. To do this, right-click the relevant constraint, select Properties and tick the primary UC option.

To add a *mandatory role constraint to a single role*, click the single role MR icon [**2**] then click the role.

To add a *uniqueness constraint to a sequence of roles*, click the general uniqueness constraint button, then hold the shift key down as you click the roles, then click the Done [✓] button. The roles may belong to one or more predicates. You can also use this for a single role if you like. If desired, you may declare the UC as primary in the constraint's properties sheet.

To add a *mandatory constraint to a sequence of roles*, click the disjunctive mandatory role constraint button, then hold the shift key down as you click the roles, then click the Done [✓] button. The roles may belong to one or more predicates. You can also use this for a single role if you like.

To add a *frequency constraint* to a sequence of roles, click the frequency constraint button [F], then hold the shift key down as you click the roles, then click the Done [✓] button. The default constraint ">=1" appears. Double-click this constraint (or right-click then select Properties) to bring up the constraint properties sheet. Then enter the required minimum/maximum frequencies. The roles may belong to one or more predicates. The same procedure applies for a single role frequency constraint.

To add an index constraint (a logical rather than conceptual issue) click the index constraint button [I], then hold the shift key down as you click the roles, then click the Done [✓] button.

You may apply a *ring constraint* to two roles of a predicate when both roles are played by the same object type. To add a ring constraint to the roles, click the ring constraint button, then hold the shift key down as you click the roles, then click the Done [✓] button. An undefined ring constraint symbol appears. Double-click this constraint (or right-click then select Properties) to bring up the constraint properties sheet. Then choose the required region from the graphic (as discussed earlier with the fact editor) to apply the relevant ring constraint(s).

To apply a *subset constraint* from role-sequence-1 to a compatible role-sequence-2, click the subset constraint button, then hold the shift key down as you click the roles in sequence-1, then click the NewSequence button, then click the click the roles in sequence-2, then click the Done [✓] button. Be sure to click the roles in the correct corresponding order. Even if there is just one role in each sequence, you have to go through the procedure as described.

To apply an *equality constraint* between role sequence 1 and role-sequence-2, click the equality constraint button, then hold the shift key down as you click the roles in sequence-1, then click the NewSequence button, then click the click the roles in sequence-2, then click the Done [✓] button. Be sure to click the roles in the correct corresponding order. Even if there is just one role in each sequence, you have to go through the procedure as described.

To apply an *exclusion constraint* over two or more compatible role sequences, click the exclusion constraint button, then hold the shift key down as you click the roles in sequence 1, then click the NewSequence button, then click the click the roles in sequence-2, … then click the click the roles in the final sequence, then click the Done [✓] button. Be sure to click the roles in the correct corresponding order. Even if there is just one role in each sequence, you have to apply the procedure as described.

Object type selector

The Object Type Selector toolbar enables you to *re-use existing object types* elsewhere on the same page or on other pages. The tool bar lists the currently defined object types, and you can drag and drop any onto the current schema page. *Always use this toolbar if you want an object type to appear more than once in the overall schema.* The system automatically adds "▲" to the top of the object type to show that it is duplicated. Do not try to enter the same object type more than once by typing it in. This toolbar may be toggled on/off by Ctrl+H or by clicking its option on the View menu.

Working with pages

Large models may be spread over several schema pages. To append a new page, choose Add from the Page menu. To insert a new page before the current page, choose Insert from the Page menu. The Page menu also lets you delete and reorder pages. Pages are always numbered, but it is a good idea to name pages to make it easier to remember their focus. To name a page, right-click on a blank area of the page, choose Properties from the popup menu, click the Page tab (if necessary) and enter the page name in the Title edit box.

At the left of the scroll bar, there are two buttons for switching between the default single page mode and multi-page mode. Mutli-page mode lets you scroll through all the pages and is convenient for viewing many pages simultaneously when the zoom factor is reduced. For multi-page mode, click the "4-corners" button second from the left [✚]. For single-page mode click the leftmost button [☐].

Validate, Build, Edit

Once you have completed at least part of your model, you can validate it by clicking the CheckDocument option from the main toolbar (enter a name when asked to save). This checks to see if the schema is valid (e.g. no contradictory constraints). More often you will go straight to the BuildDictionary option (from the main toolbar or from the Database menu): this first validates then maps the ORM model to a logical model.

If you want to look at or edit the logical model, choose EditDictionary (from the main toolbar or from the Database menu). This does the build then opens the logical model for you to work with. The table icons appear in a Tables/Entities window. Select the tables you want to view and drag them onto the logical document pane.

The automatically generated names are often non-ideal. You can edit them by double-clicking them, and entering your changes in the Column Properties window. Like some other windows, this has a thumb-tack which may be toggled on to keep the window present as you move from one column to another. Thumb tacks can speed up editing tasks considerably.

You can view the logical model in pure relational notation (my preference) or in IDEF1X notation. To modify the view, right-click on a blank area of the logical document page, choose DocumentSettings, and make your choices.

To see the conceptual origin of a logical column, right-click on the column then select Go to Definition from the popup sheet. The conceptual schema will be displayed with a red arrow pointing to the relevant predicate that mapped to the column.

Use the pull down menu from the Window option on the main menu to switch your viewing windows. The .imo option indicates the ORM window, the .imd option indicates the dictionary (logical) window, and the Output option indicates the window for listing messages.

Generate

Once you have edited the logical model, you can generate the internal schema for a selected target DBMS by clicking the GenerateDatabase icon from the main toolbar (or selecting Generate Database from the Database menu). Follow the steps in the generate wizard, choose your driver (e.g. Microsoft SQL Server), enter a database name (e.g. mydb), accept defaults for the next screen, and choose Yes to view the generated DDL script. You can save the DDL script as a textfile by selecting Save from the File menu in Notepad. For printing you may need to first select PageSetup from the File menu and change the page size to A4.

Printing

The US version of VisioModeler defaults to Letter size for pages. If you use A4 paper instead, use PageSetup if needed to change the page size to A4 and set this as the default. You may need to do this for each kind of document you want to use.

Save your work

Once you start work on the actual application, at least at the end of each session you should save your work by saving it (⊟ icon) then copying at least your .imo file(s) to a 9 cm floppy disk. Insert your disk in drive a:, select your file in Explorer, Ctrl+C to copy it to the clipboard, select drive a:, and Ctrl+V to paste it to your floppy disk (or use Ctrl-drag to copy it). If you haven't already done so, create your own Models folder for storing your models. The Samples folder contains a number of sample models for you to play with. Each time you open VisioModeler, it opens the models that that were last worked on. If you are working on somebody else's PC (e.g. in a university laboratory) then after copying your work you should delete it from the PC (e.g. select the file in Explorer and hit the Delete key).

Exit

Exit VisioModeler by clicking its ⊠ button at the top right corner.

Relational mapping,

Controlling primary key choice

If you have a ternary or longer predicate with overlapping uniqueness constraints, you may want to pick the primary key yourself rather than having VisioModeler pick its own choice. The same applies if you have a nested or co-referenced entity type with more than one possible reference scheme. In this case, right-click the uniqueness constraint you want to be primary and tick the relevant box in the properties sheet that appears (see below). Your choice is then indicated by a "P" symbol on the ORM schema.
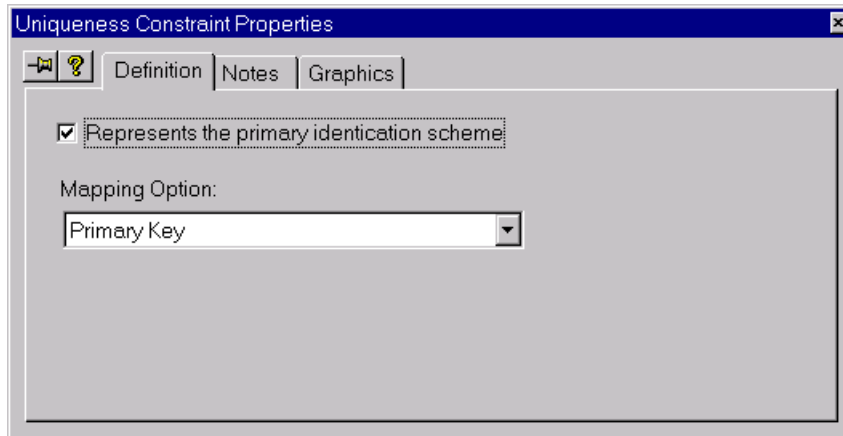
Controlling column names before mapping

Read the previous handout to see how to enter and validate an ORM model. You can use the EditDictionary button to map an ORM model to a relational (logical) model, which is stored as a dictionary file (with file extension ".imd" for InfoModeler Dictionary). Before you do this however, you can make some choices to control how VisioModeler generates the names for the tables and columns in the logical model.

For each entity type with a reference mode, double-click it to open its Object Type Properties sheet, open the RefMode pane and choose which of the four naming options are appropriate (see below figure).

For reference modes like "name" or "nr" it is usually best to choose Add to Object Type name. This causes the reference mode to be appended to the entity type name when forming the column name. By default, an underscore separator is added. For example, Country(name) ➞ Country_name, Employee(nr) ➞ Employee_nr. In cases where two roles played by the same object type map to the same table, extra text is added to the column names to distinguish them.

If you don't want the reference mode to be used at all, choose Do not use for naming. If the object type plays only one role, its name is used as the column name. For example, Status(code) ➞ Status, Rating(nr) ➞ Rating, Date(mdy) ➞ Date.

If you want the reference mode itself to be used as the column name, choose the option Use as Column name. For example, Employee(empnr) ➞ empnr. I rarely use this option, even


though it is the default. You can change the default by right clicking on some blank space in the ORM document to open the Document/Page menu and choosing Document Options. Go to the Miscellaneous pane (see next figure) and make your choice.
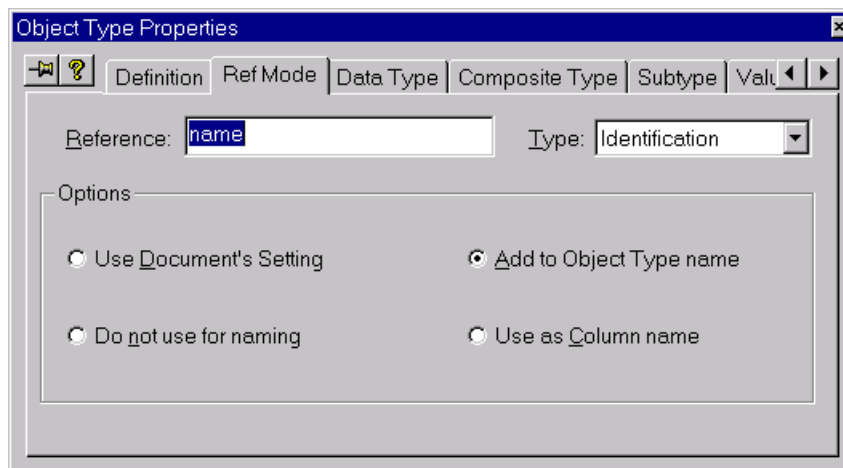
If, like me, you usually don't like underscores, you can turn them off by selecting None for the Spacing Character. With this choice, the Add to Object Type name option appends without the underscore, e.g. Country(name) ➡ Countryname.
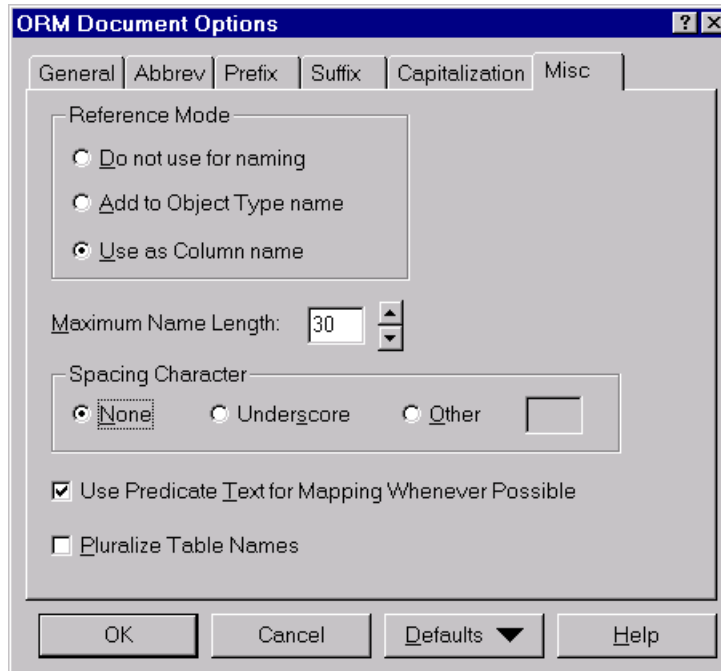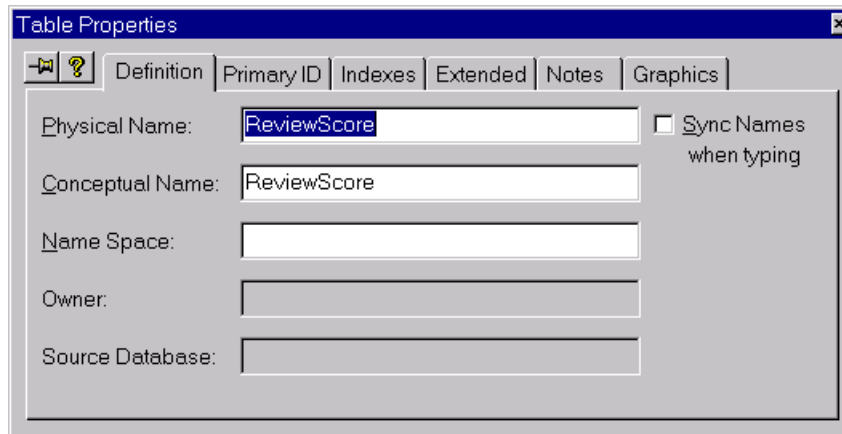

For cases where two or more roles played by the same object type are mapped to the same table, VisioModeler adds distinguishing text from the predicate to provide different column names. However these generated names are often awkward. You can gain a certain amount of control over this by adding role names to the predicate. Double-click the predicate to open its properties sheet and go to the Readings pane. Then enter role names as appropriate. These will then be used either wholly or partly for the column names.
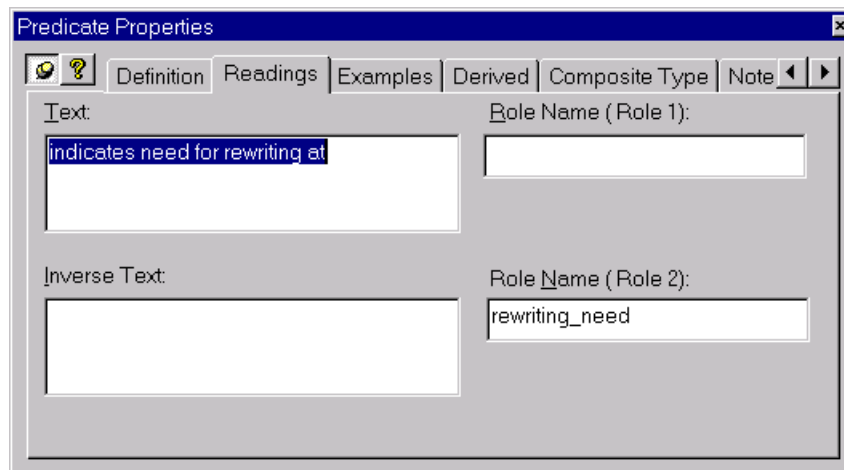
Controlling table names, the order and names of columns, and adding notes after mapping

After you have mapped an ORM schema, there will normally be further refinements you want to make to the tables and columns. If you used EditDictionary for the mapping, drag the relevant tables out onto the logical document window. You may change the order of the columns in any table simply by dragging the columns to the position you want with the mouse.

To rename a table, double-click it to open its properties sheet, then go to the Definition pane and enter the new name (see below). Now go to the Notes pane and add a meaningful description.

Table Properties

| | | Definition | Primary ID | Indexes | Extended | Notes | Graphics | |

Physical Name: `ReviewScore` ☐ Sync Names when typing

Conceptual Name: `ReviewScore`

Name Space: _____

Owner: _____

Source Database: _____

To rename a column, double-click it to open its properties sheet, then go to the Definition pane and enter the new name (see below). Now go to the Notes pane and add a meaningful description.

Column Properties

| | | Definition | Data Type | Value | Extended | Notes | |

Physical Name: `exitCountry` ☐ Sync Names when typing

Conceptual Name: `Countryname`

☐ Allow NULL Values
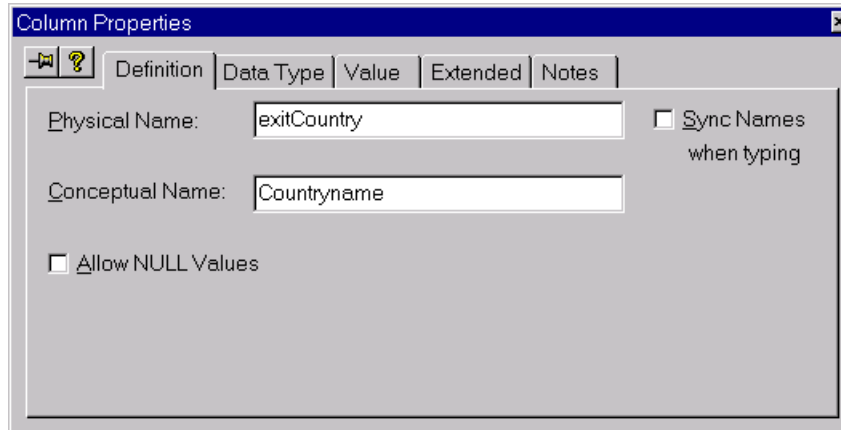
After you've made changes like this to the logical model, it's best to save your changes before making any changes to the conceptual model. Otherwise, when you later save your new conceptual model you will be asked whether you want to migrate the unsaved logical model changes at that point. If you say yes, your logical changes are saved and migrated back to the conceptual model, but you lose any subsequent changes you had made to the conceptual model. It's good practice to always save changes to your current document (ORM or Logical) before moving to another document.

Table reports

VisioModeler provides extensive reporting features for all your documents, but we consider only basic table reports here. To create a new report, select the New Report option from the File menu. A wizard guides you through the next steps. Select Table Report, then Finish. This opens up the report window. Now go to the Preferences pane and select Relational notation and Physical names (see next figure).
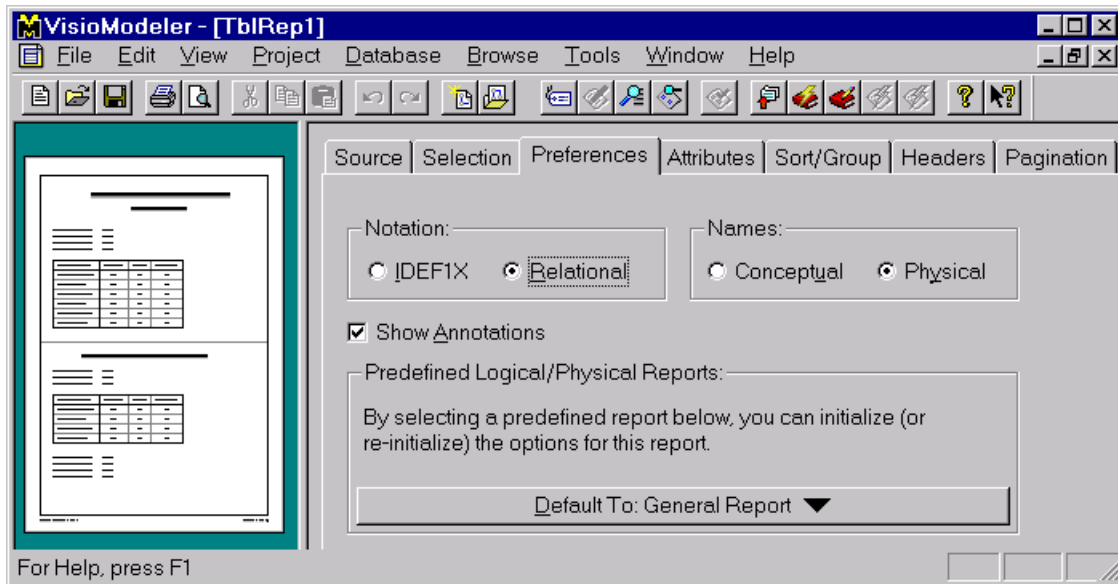
Now go to the Attributes pane. Attributes for the chosen category are listed with check boxes so you can select or de-select them as desired. All options are selected for Database Attributes. Leave this as is, change the category to Table Attributes and select just the five options shown below (or whatever you wish).
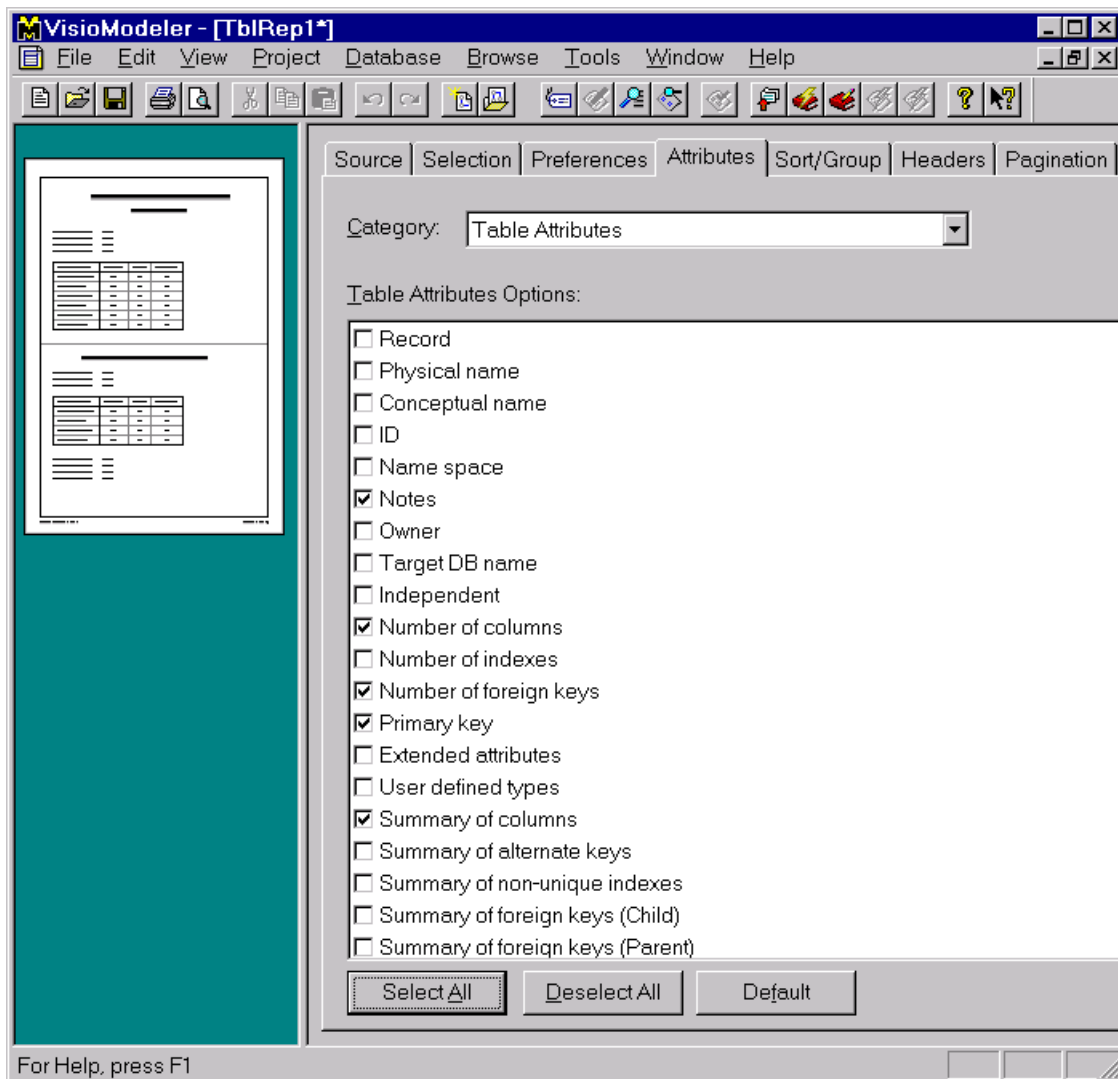
Now choose the Column Attributes category, and select just the Notes option (see below). This will arrange for the notes you have added for each column to be printed.
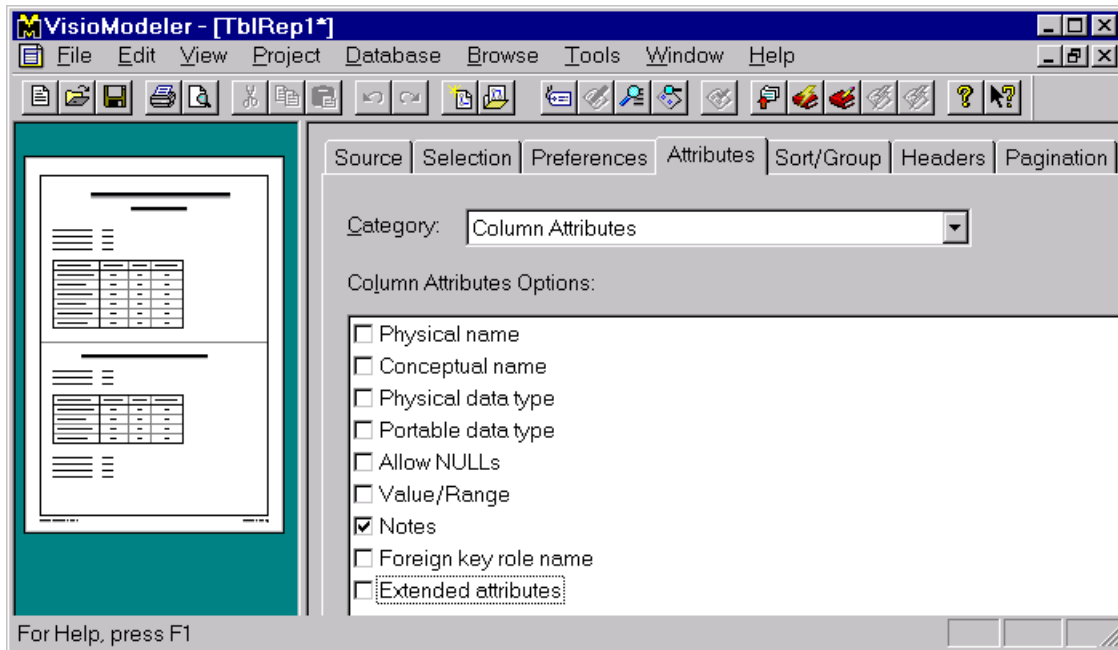
Now choose the Index Attributes category and hit the Deselect All button (so none are ticked). Now choose the Foreign Key Attributes category and Deselect All.

Now go to the Pagination pane, and change the setting of Group Separator to Double Line (see below). This causes each subsequent table to be separated from the previous one by a double line, rather than starting on a new page (the default). This choice is mainly aimed at saving paper.
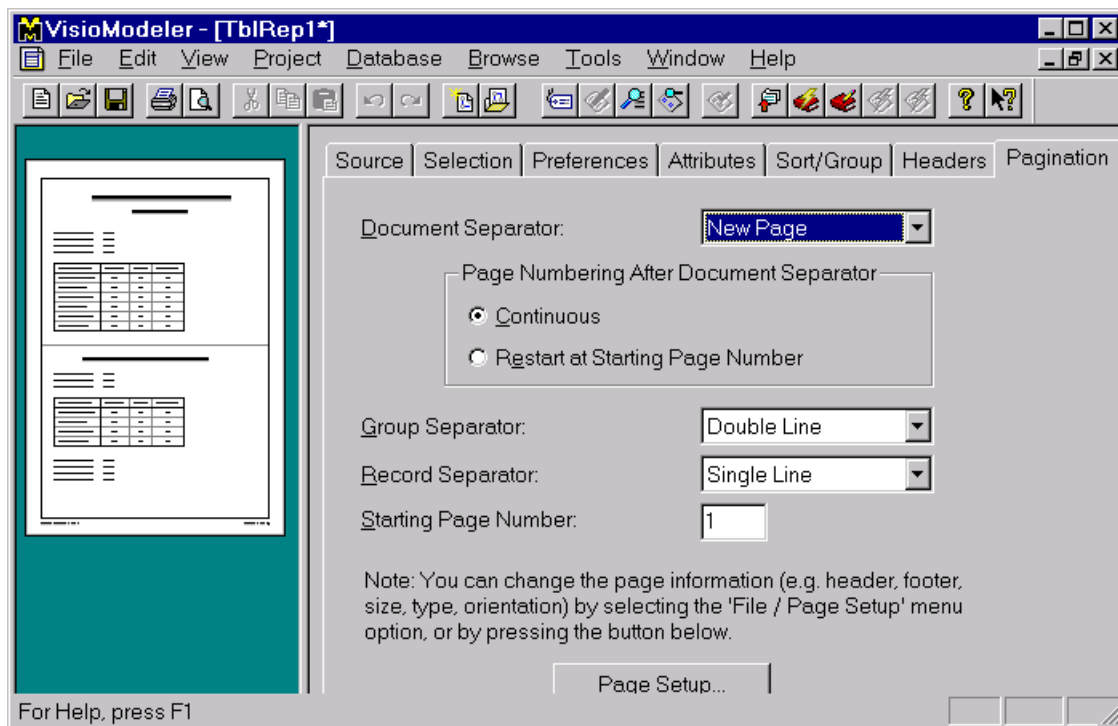
If you haven't already done so, ensure that the relevant logical document is selected in the Source pane. An example is shown in the next figure.

Now use the Print Preview button to preview how the printout will appear (see below). You can move between pages and zoom in and out using the buttons in the tool bar. VisioModeler should automatically select the default printer settings, but if needed you can reset the page size (e.g. to Letter or to A4). If necessary, close the preview window (hit the Close button) and make changes to the report settings until the preview shows what you want. Then hit the Print button to print the report.

You can save your report, by hitting the Save button on VisioModeler's Main toolbar and giving the file a name. Report files are saved with the extension ".irp" (VisioModeler RePort). When finished, hit the ⊠ button on the current window. After performing any other necessary saves, hit the outermost ⊠ to exit VisioModeler.