

Building a database application using OOo Base

REACHING BASE

www.sxc.hu

If you need a quick and easy solution, try building your own database application in OpenOffice.org 2.0.

BY DMITRI POPOV

OpenOffice.org 2.0 features a new database module based on HyperSonic SQL, a Java-based relational database engine. You can use OOo Base to build anything from a simple recipe collection to a powerful asset management solution. Base is a powerful tool, but for those not familiar with its features, developing even a simple database solution can be a bit over-

whelming. This article guides you through the steps of setting up a simple business database application using Base.

Planning the Database

A Relational Database Management System (RDBMS) typically consists of four parts:

- tables
- queries
- forms
- reports

Data within a database is stored in tables. Queries are used to extract, view, and manipulate the data. Queries can draw together data from many tables, and

that data is then available for forms and reports. Forms allow you to view and edit the data in a table. Think of a form as a GUI for the database. Reports are used to produce formatted output from the table, usually for printing.

As you'll learn in this article, you can break the task of creating a database application into the smaller tasks of setting up tables, queries, and forms. Reports are also important for some applications, however, we will not delve into the subtleties of reports for this article.

Before you launch Base, it is a good idea to create a database model that will

THE AUTHOR
Dmitri Popov holds a degree in Russian language and computer linguistics. He has been working as a technical translator and freelancer contributor for several years. He has published over 500 articles covering productivity software, mobile computing, web applications, and other computer-related topics. His articles have appeared in Danish, British, US, and Russian magazines and web sites.

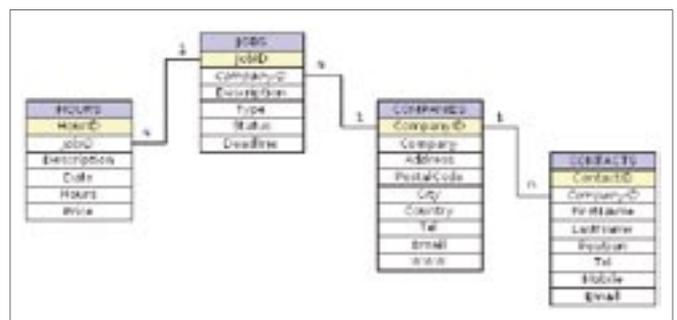


Figure 1: The database model includes tables, fields, and relations.

help you visualize the structure of the application. You can either draw the model by hand, or you can use a tool like OpenOffice.org Draw. Draw includes some diagramming tools that are helpful for creating this kind of database model.

The example application in this article is a simple database that tracks jobs, work hours, and clients for a freelance contractor worker. Figure 1 shows the model for our simple example application. As you can see, the structure consists of four small tables. Each table consists of a series of records depicting a COMPANY, CONTACT, or JOB, or the HOURS worked on the job. The records are linked through a series of predefined relationships. The relationships let one table reference data in another table. In Figure 1, for instance, each job is associated with a company for whom the services are performed. This structure supports the situation in which a contractor may have several contracts (jobs) for a single company or may have several contacts within the company.

Through the series of relationships shown in Figure 1, one company can be associated with many jobs and many contacts, and each job can have many time entries. All the relationships in Figure 1 are *one-to-many* relationships. A one-to-many relationship is noted with a 1 (one) on the side of the unique record and an n (many) on the opposite side.

Every table must have a primary key (shown in yellow in Figure 1). The primary key uniquely identifies a record. A table may also contain a foreign key – a field that references a primary key in another table. In Figure 1, the foreign keys are shown in italics. Note that a foreign key is on the “many” side of each of the one-to-many relationships.

Creating Tables

Once you have sketched out the design, you can begin developing the actual database. Launch OpenOffice.org 2.0 Base and choose *File | New | Database*. Use the Database Wizard to create and save

a database file. When you click the *Finish* button in the Database Wizard, OOo Base opens the main window. The main window is the central working zone where you will create your database (Figure 2).

The first thing you need to do is create the tables and populate them with fields. Choose *Tables* in the *Database* pane, and click on the *Create Table* option in Design View. This option will create a new empty table, which we will use to store job data. Create a *JobID* field by typing its name in the *Field Name* column, and set its type to *Integer* [INTEGER].

OOo Base offers a wide range of field types. Numerical types include *INTEGER* (for integers) and *FLOAT* (for floating point numbers). *VARCHAR* stores variable-length strings; the width can be specified in the *Field Properties* pane. *DATE* stores a date, and *TIME* stores a time.

To set the *JobID* field as the primary key, right-click on the record pointer (the green triangle) and select

Primary Key from the context menu (Figure 3). The primary key has to meet two requirements: it must not be empty, and it must have a unique value. OOo Base makes it easy to assign this property to the primary field key. Select *Yes* from the *AutoValue* list in the *Field Properties* pane. This will ensure that every time you create a new record, the system will insert a unique value (in our case, it’s a number) in the *JobID* field.

Add the rest of the fields in the *JOBS* table, and set their types. Before you save the table, make sure that the date format of the *Deadline* field is configured properly. Select the *Deadline* field, and in the *Field Properties* pane, click on the button next to the *Format example* field. Select the desired format and language and click *OK*. Save the table (*File | Save*,

or *Ctrl + S*), and when prompted, give the table the name *JOBS*.

Now that you know how to create tables and fields, you can add the *HOURS*, *COMPANIES*, and *CONTACTS* tables. Just remember to set the right format for the *Date*, *TimeUsed*, and *Price* fields in the *HOURS* table.

With the *Relation Design* tools included with Base, creating relations between tables is as easy as drawing lines between primary and foreign keys. To define relations between the four tables, choose *Tools | Relationships*, and use the *Add Table* dialog to add the tables.

To establish a relation between the *COMPANIES* and *CONTACTS* tables, select the *COMPANIES.CompanyID* field, click and hold the left mouse button, then drag a relation onto the *CONTACTS.CompanyID* field and release the button. The next step is to define the relation’s properties. Let’s say you have a company in your database that is no longer your customer. If you decide to delete it from your database, the system has to know what to do with the orphaned contact records. Normally, you’d want to delete them as well.

Double-click on the relation between *COMPANIES* and *CONTACTS*, and in the *Delete Options* column, select the *Delete cascade* option and click *OK*.

Using the same technique, define the relations between the *COMPANIES* and *JOBS* tables. Make sure the delete option of the relation is set to *No action*. Otherwise, deleting a job will delete the related company. Finally, define a relation between *JOBS* and *HOURS* tables, where the delete option is set to *Delete cascade*. This ensures that if you delete a job, you won’t have orphaned time entries floating in your database. When the relations are defined, save them and close the *Relation Design* dialog window.

Creating Queries

Now you have tables, and gradually you will fill them with data. To view and manipulate the data, you have to create



Figure 3: Every table needs a primary key.



Figure 2: Base’s main window is the starting point for defining your database.

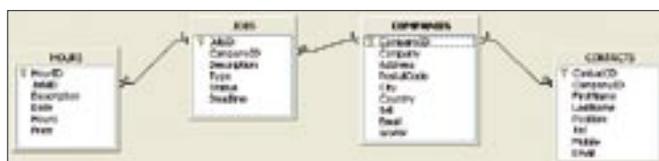


Figure 4: Defining relationships in OOo Base.

Advertisement

queries. OOo Base allows you to create queries using either the Query Wizard or the Design View. Users familiar with SQL can also use the built-in SQL editor. In most cases, though, you may want to use the Design View, which allows you to construct queries using a graphical user interface. Let's say you want to create a query that allows you to extract relevant data about a job, such as the job number, customer, job description, status, type, and deadline. Click on the *Create Query in Design View* link in the *Tables* section. This opens the Query design window and the Add Tables dialog. Add the *JOBS* and *COMPANIES* tables to the query using the *Add* button, and click *Close*.

All you have to do now is add fields to the query. You can add fields either by choosing the fields from a drop-down list in the *Field* cells, or by dragging fields from the tables to the *Field* cells.

By default, the query will use the fields' names as their labels, but you have to change that to make your query more readable. For example, you can display the *JobID* field under the name *Job no.* by typing the latter in the *Alias* cell. To save the finished query, choose *File Save* and give the query a name. The new query appears in the Query section, and you can launch it by double-clicking on it.

Building Forms

To make your database application complete, you need to build a form-based interface. Let's say you want to create a form that you can use to display, edit, and add job records. Switch to the *Forms* section by pressing the *Forms* button, then click on *Create Form* in Design View. This will open a blank window. Click on the *Form Navigator* button on the Form Design toolbar. In the Form Navigator, right-click on the *Forms* folder and choose *New | Form*. Give the form a descriptive name (for example, *Job*), then right-click on it and select *Properties*. Under the *Data* tab, select *Table* from the Content Type list and *JOBS* from the *Content* list.

To add the Description text field control (which contains the job description), click on the Text box button on the Form Controls toolbar, and draw a text box in the form. Double-click on the text box. Under the *Data* tab, select *Description*

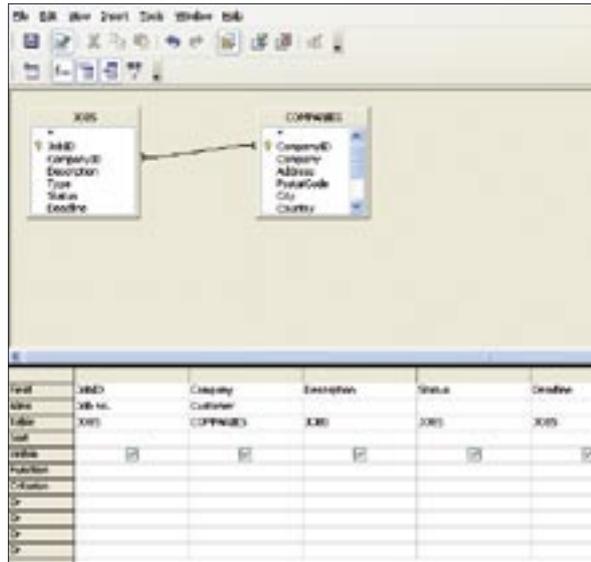


Figure 5: Base provides a graphical interface for creating a query.

from the *Data* field list. In a similar manner, you can add other fields like *Status*, *Type*, and *Deadline*. Using the Properties window, you can add short help texts to the controls. To do this, click on the *General* tab, and enter the help text in the *Help Text* field.

This new form allows you to add, view, and edit data in the *JOBS* table, but you can also include data from other tables. For example, you may want to view and register the time spent on the currently viewed job. You can do this using a subform. As the name implies, a subform is just another form inserted in the main form.

In the Form Navigator window, click on the *Job* form and select *New | Form*. Give the subform a descriptive name (for example, *Hours*), then right-click on it and select *Properties*. Under the *Data* tab, select *Table* from the Content Type list, and *HOURS* from the Content list. You can then add form controls to the subform. Since you can have multiple records in the *HOURS* table that are linked to a particular job, you may want to view them as a table in the Job form. For this purpose, OOo Base conveniently offers the Table Control. Make sure the *Hours* subform is selected in the Form Navigator, click on the Table Control button, and draw a table. When the Table Element Wizard appears, select the fields you want and press *Finish*.

When you are done creating the form, save it and close the Form Designer. Double-click on the form to open and

use it. To browse, add, and edit records, use the Form Navigation toolbar (*View | Toolbars | Form Navigation*). You can also use it to sort and filter the records.

Importing data

Using forms, you can easily create new jobs and customers, and add contacts. But what if you already have this data in a spreadsheet? In many situations, you just want to import the existing data into your database. For instance, say you have a Calc file containing a list of contacts,

and you want to import it into the *CONTACTS* table.

Since properly formatted source data can greatly simplify the import procedure, it's worth spending time on some preparatory work. First of all, you need to tweak the spreadsheet's structure so it becomes identical to the structure of the *CONTACTS* table. Make sure the first row contains field names identical to those in the database table. Ideally, they should also be in the same order. Ensure that all the data bits are placed in the proper columns.

Now select the data in the sheet and copy it into the Clipboard (*Edit | Copy* or *Ctrl + C*). Open the database, click on the *Tables* button, and select the *CONTACTS* table. Right-click on it, and choose *Paste*. In the Copy table dialog, type the exact name of the target table into the *Table name* field, and select the *Attach data* option.

In the Assign Columns dialog, check whether all fields in the Source table window are selected and aligned with the respective fields in the Target table window. To align the fields, use the blue up and down arrows. Finally, click *Create* to import the data.

Conclusion

This article should get you started with exploring the new OpenOffice.org Base database module. Once you get used to Base, you can easily apply these techniques to your own database applications. ■

Advertisement