

PAPER • OPEN ACCESS

A Remote Registration Based on MIDAS

To cite this article: Xin JIN 2017 *IOP Conf. Ser.: Earth Environ. Sci.* **61** 012154

View the [article online](#) for updates and enhancements.

You may also like

- [Deep learning in medical image registration: a review](#)
Yabo Fu, Yang Lei, Tonghe Wang et al.
- [Deformable image registration for cone-beam CT guided transoral robotic base-of-tongue surgery](#)
S Reangamornrat, W P Liu, A S Wang et al.
- [Non-rigid registration of medical images based on estimation of deformation states](#)
Bahram Marami, Shahin Sirouspour and David W Capson



ECS
The
Electrochemical
Society
Advancing solid state &
electrochemical science & technology

DISCOVER
how sustainability
intersects with
electrochemistry & solid
state science research

A Remote Registration Based on MIDAS

JIN Xin

Department of Computer Wuhan Polytechnic (Wuhan, China)
38390681@qq.com

Abstract. We often need for software registration to protect the interests of the software developers. This article narrated one kind of software long-distance registration technology. The registration method is: place the registration information in a database table, after the procedure starts in check table registration information, if it has registered then the procedure may the normal operation; Otherwise, the customer must input the sequence number and registers through the network on the long-distance server. If it registers successfully, then records the registration information in the database table. This remote registration method can protect the rights of software developers.

1. The realization principle The first section in your paper

MIDAS (Multi-tier-Distributed Application Services Suite) is a multi-tier distributed data application technology, this technology is based on the DataSnap database. Programmer do not need to write a special program code for database access, the programmer can more easily and conveniently develop database application system based on C/S. The user of Delphi development platform can easily implement software registration program. The principle can be seen in Figure 1.

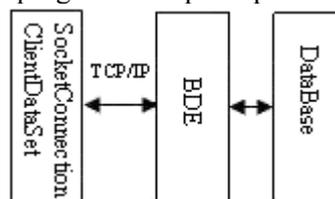


Figure 1. The principle of DataSnap database application system

It can be seen from figure 1, the application software using “BDE SocketConnection” to connections, which need to connect through TCP/IP protocol. “BDE Socket Connection” links to Database by SQL. It is easy to implement in Delphi. We are to realize the network software registration using the multilayer distributed database principle.

2. Steps to realize remote registration

After the customer application installed on the local host, will automatically put the table file “register.db” corresponding to the installation folder. Table register.db contains a field named “registered”, the data_field type is Logical. In the remote login server, we install a table named server.db, server.db contains three fields: the “serialno”, the “reged_num” and the total_num.

The realization of software DataSnap based on remote registration are as follows:

(1) After the client program start, We open the local host table file “c:\windows\ register.db”(for secrecy, in reality you can use file irrelevant, the extension does not need to be “db”, the file is hidden, placed in the irrelevant folder). Read the field reged_num in the file “server.db”. If the “registered”



value is true, it indicating that the user is a valid user, then it display module interface, hide the registration interface, you can use the client application.

(2) If the "registered" is the value of the false, it means the user has not registered. Program hiding function module interface, it displays the registration interface, the user must enter the serial number. The client software to the input serial number and the remote server in server.db each record serialno comparison. If the file "server.db" does not exist in the serial number of the input, this means that the client software is illegal. The Client software will be forcibly end run.

(3) If server.db exists in the input serial number. Is the serial number corresponding to the serialno and total_num. If the reged_num is equal to total_num, registered failure, end the client application. If the serialno we input is in the file "server.db". Put the serial number corresponding to the reged_num plus one, and the client in the file register.db "registered" set to true, prompt software registered successful. It display function module interface, hide the registration screen. At this point, based on DataSnap software registration process is complete.

3. Specific implementation based on datasnap software registration

3.1 Establish a remote login information table file "server.db"

We establish the remote server registration information table file "server.db". The table has three fields: serialno、 reged_num and total_num. We save the file to the "d:\ remote login server \database" folder. The table file as shown in Figure 2.



| server | Serino | Server | Reged_num | Total_num |
|--------|--------|--------|-----------|-----------|
| 1 | 123456 | 1.00 | 0.00 | 1.00 |
| 2 | abcdef | 2.00 | 0.00 | 10.00 |
| 3 | hello | 3.00 | 0.00 | 20.00 |

Figure 2. The table file of server.db

The meaning of each field is as the follows:

Serialno: The services provided to customers, the strings for the use of register. If there is a serialno in the register file, the user is allowed to register. If the serialno of input is not exists in the register file, it means the user is illegal, will be prohibited to register the software.

reged_num: The serialno corresponding to the number of registered.

total_num: This software allows the serialno of the registration number.

In order to make registration more powerful software, different serialno allows registration number may vary. When the reged_num is less than the total_num, it allows the serialno to register. If the reged_num is equal to total_num, this serialno is not allowed to register, customers must buy new serialno.

A serialno is allowed to register multiple; this is in order to meet the needs of users to install software on multiple machines.

3.2 Establish the remote registered server

The remote registered server is connected to the table file "server.db"; provide the software registration service for customers. The establishment of the registration server comprises the following steps:

(1) We using Delphi development platform, Establishment of an application. Add four controls on the form: Label、 Table、 DataSource、 DBGrid and DBNavigator. The main properties of these controls such as table 1. Save the form and project, named "zhucefwqunit1.pas" and "zcfwq.dpr". Unit file and project files are saved to had built in folder "d:\ remote login server \database".

Table 1. The properties of the controls

| control | property | property value |
|--------------|--------------|-----------------------------------|
| Table1 | DataBaseName | d:\ remote login server \database |
| | TableName | server.db |
| | Active | True |
| DataSource1 | DataSet | Table1 |
| DBGrid1 | DataSource | DataSource1 |
| DBNavigator1 | DataSource | DataSource1 |

(2) We establishment of remote data module. In accordance with the order of execution of the menu command: [File]→[New]→[Others], open the New Item dialog, select “Multitier” page, select “Remote Data Module”, save the unit named "socketzc.pas", save the class named "socketzhuce". The saving location is “d:\ remote login server \database”.

(3) We add a “table” control and a “DataSetProvider” control for the remote data module. The main properties of these controls such as table 2.

Table 2. Remote data module control's properties

| control | property | property value |
|------------------|--------------|-----------------------------------|
| Table1 | DataBaseName | d:\ remote login server \database |
| | TableName | server.db |
| | Active | true |
| DataSetProvider1 | DataSet | Table1 |

Explain: The DataBaseName property Table1 must be set to an absolute path.

(4) After we run Socket proxy server(in the folder "Borland\Delphi7\Bin"). The file name is “scktsrvr.exe”, this can be seen when the server has been running in the lower right corner of the icon windows.

(5) After we run the registration server, the registration server is registered in the proxy server.

So far, the registration server has been running normally, as shown in Figure 3 and 4.

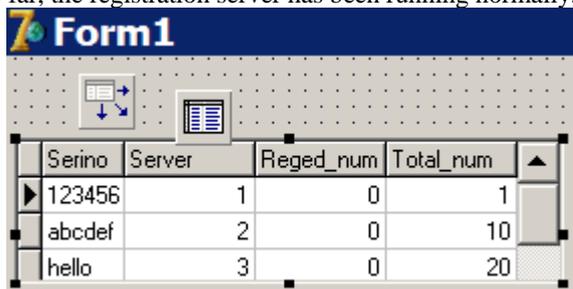


Figure 3. the remote register server

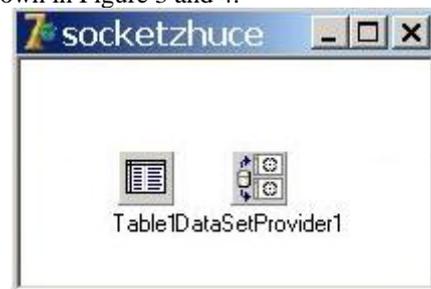


Figure 4. the remote data module

3.3 The establishment of the client application

We first create a file “register.db” on the client application, this table contains only one field "registered", the data type is "logical", the establishment of the client application, this application contains two forms named “form1” and “form2”, the user can use the form1 to software registration, form2 is a software function area. Add controls “SocketConnection”, “ClientDataset”, “Table” in form Form1. The table of “register.db” is shown in Figure 5, and the form1 is shown in Figure 6.

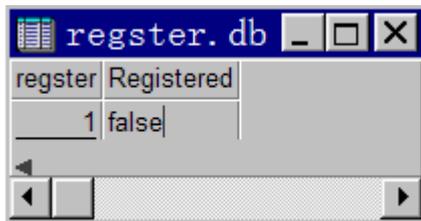


Figure 5. The table of “register.db”



Figure 6. The registration form form1

Some important property of the controls is in Table 3.

Table 3. Some important property of the controls

| control | property | property value |
|-------------------|--------------|-------------------|
| SocketConnection1 | address | 127.0.0.1 |
| | Port | 211 |
| | ServerName | zcfwq.socketzhuce |
| | Connected | false |
| ClientDataSet1 | RemoteServer | SocketConnection1 |
| | ProviderName | DataSetProvider1 |
| | Active | false |

Explain: Control SocketConnection is responsible for connecting to the remote server; Control ClientDataSet1 is responsible for connecting to the remote table file in server.db by SocketConnection1. The SocketConnection connected property and the ClientDataSet1's Active property is set to false. In the implementation of the registration operation and set them to true. The properties of Table do not need to set, its properties set to complete in the program through the code.

We save the program to the “d:\local”. In the “procedure TForm1.FormActivate(Sender: TObject)” event process to determine whether the client application has registered, If no registration, registration is required before it can be normal use.

The key algorithm is as follows:

```

procedure TForm1.FormActivate(Sender: TObject);
var
  path: array[0..MAX_PATH] of Char;
begin
  GetWindowsDirectory(path, MAX_PATH);
  table1.DatabaseName:=path;
  table1.TableName:='server.db';
  table1.Active:=true;
  table1.First;
  if table1.FieldByName(' You have successfully registered.').AsBoolean=false
  then begin showMessage(' You have not registered'); form1.Show; form2.hide; end
  else begin form2.show; form1.Hide; end;
end;

```

In the button1 write event process key code is as follows:

```

procedure TForm1.Button1Click(Sender: TObject);
begin
  socketconnection1.Connected:=true;
  clientdataset1.Active:=true;
  clientdataset1.First;
  if clientdataset1.Locate(' serialno ',edit1.Text,[loPartialKey]) then
  if clientdataset1.FieldByName('Reged_num').AsInteger

```

```
<clientdataset1.FieldByName('Total_num').AsInteger then
begin
clientdataset1.Edit;
clientdataset1.FieldByName(' Total_num ').AsInteger:=clientdataset1.FieldByName
(' Total_num ').AsInteger+1;
clientdataset1.Post;
clientdataset1.ApplyUpdates(2);
table1.Edit;
table1.FieldByName(' registered ').AsBoolean:=true;
table1.Post;
showmessage(' Congratulations on your successful registration ');
form1.Hide;
form2.Show;
end else
begin
showmessage(' The serial number of the registration number is full, Please buy a serial number ');
application.Terminate;
end
else begin
showmessage(' You are an illegal user ');
application.Terminate;
end;
end;
```

So far, the remote registry software has been implemented.

4. Conclusion

Based on the remote registry DataSnap software can solve a sequence number is arranged in a multi machine problems. Author throws away a brick in order to get a gem, hope that readers can infer other things from one fact. For example: to add a field in the table server.db, this field represents a registration allows the number of runs, The field "registered" of local table "register.db" changed into "remaining number". In this way, the registration system function will be stronger.

Acknowledgment

This work is supported by 2014 year Hubei Province Department of education science research project, the project name is "Design and implementation of online examination platform").

References

- [1] HOU taiping,TONG aihong.Delphi Database programming[M].Beijing:Tsinghua University press.
- [2] Product R & D Center of FEISI Science and technology. Delphi 6 distributed development[M].Beijing: Publishing House of electronics industry.
- [3] JING Hai.The Delphi network programming[M].Beijing: China Railway Publishing House.
- [4] Ray Liscbner.Delphi technical manuals[M].Bejing:China electric power press.
- [5] LI Huan,Gaoling,LIU ling,XIN Bin. Improved algorithm for generating random CAPTCHA[J]. Journal of Computer Applications. Vol.30 No.6 June 2010:1501-1504.