Available online at http://www.mecs-press.net/ijeme

Development of GIS Application System on OCCI and MapX

Li Guangshi[a],*

[a] Computer Center of Anshan Normal University, Anshan, Liaoning, China

Abstract

Exploring a method of developing application system which can access and manipulate a mass of spatial data effectively under integration storage mode of GIS data in Oracle Spatial, is the research hot spot. This paper defines more compact, robust and reliable class modules for accessing spatial data, carries on the visualization processing of spatial data using MapX, and gives a method for enhancing the ability to manipulate spatial data in GIS system through using OCCI to call spatial operators in Oracle Spatial, which unites advantages of OCCI, MapX and Spatial to develop GIS application system fully.

Index Terms: Spatial Data; Oracle Spatial; OCCI; MapX

1. Introduction

With the application of Geographic Information System (GIS) in all areas of society, cubage of spatial data in the GIS increasingly increases. Using Oracle Spatial spatial database technology to store and manage spatial data and attribute data in GIS is favored by GIS developers increasingly. In this new integration storage mode, how to give full play to the advantages of object-relational model to more conveniently and efficiently manage massive spatial data in Oracle database becomes key issues that are faced by developers of GIS. This paper attempts to develop a GIS application based on MapX component and Oracle C++ Call Interface (OCCI) in VC + + environment.

2. The Advantage of Developing GIS Based on the OCCI and MapX

2.1. Advantages of the OCCI Interface

Currently, there are many C + + interfaces that can access Oracle Database , such as ODBC and ADO that are simple to use but correspondingly slow and limited by the operating system（and they do not support access Oracles' data object types）, the Pro C/C++ that is more traditional, but inconvenient debugging interface, the OCI is low-level interface which has the fastest access speed, the most powerful security and performance, but

---

* Corresponding author:
E-mail address: [a]wslgs@netease.com

it's using is extremely complicated, etc[1-3]. Due to spatial data in Oracle Spatial are stored in the form of object, and the mass storage, in view of this situation, an effective interface which can be called conveniently and close to bottom layer is needed to use.

The Oracle C++ Call Interface (OCCI) is an application programming interface (API) that allows applications written in C++ to interact with Oracle database servers. OCCI directly implements the object-level encapsulation based on the OCI, which has the advantage of flexibility, high efficiency, succinct code , easy understanding and easy maintaining and provides a library of standard database access and retrieval functions in the form of a dynamic runtime library (OCCI classes) that can be linked in a C++ application at runtime. OCCI enables C++ programs to utilize the full range of Oracle database operations, including SQL statement processing, object manipulation and executing PL/SQL, it provides a navigational interface to manipulate database objects of user-defined types as C++ class instances[4,5]. Therefore, it has more and more customers.

2.2. The Advantages that OCCI Combines with MapX

Although the client-side can easily read, write, query GIS data in Oracle database by OCCI applications, it oneself actually does not have the visualization function for the GIS spatial data, while this task can be achieved by MapX . MapX is an ActiveX control which has powerful map functions, produced by Mapinfo company. It provides a highly visualization method to display and analyze location-based data and can support most standard visual development environment[6]. Using MapX to develop GIS system has the predominance of short cycle, low cost. This way accords   the current component development trends of GIS application system[7] . At the same time, because Oracle Spatial provides rich spatial operation functions and procedures[8]， the OCCI can also call the powerful functions and procedures in Oracle Spatial to make up for MapX's functional deficiencies in spatial query and spatial analysis, to efficiently implement various spatial query and complex spatial analysis.

3. The Key Technologies and Solutions of Developing Application System Based on OCCI and MapX

This paper uses Oracle 10g Spatial as the spatial database and develops GIS application system based on the OCCI interface and MapX in VC++ environment. The following content is detailed description about key technologies in the development.

3.1. The Methods of OCCI Accessing Oracle Spatial

Though the OCCI encapsulates the OCI in the form of classes, in GIS applications, the developers still hope the GIS system is divided into several subsystems, it can provides a simple and unified interface for each subsystem to reduce the direct dependence between external program and function realization . So, the OCCI can be further encapsulated to produce more robust, more reliable and easier to maintain the class module [9]. This paper mainly defines two classes for interacting with Oracle spatial database server: User_Defined_Geometry and geoProc. The process that accesses the Oracle spatial database through OCCI is shown in Fig. 1.
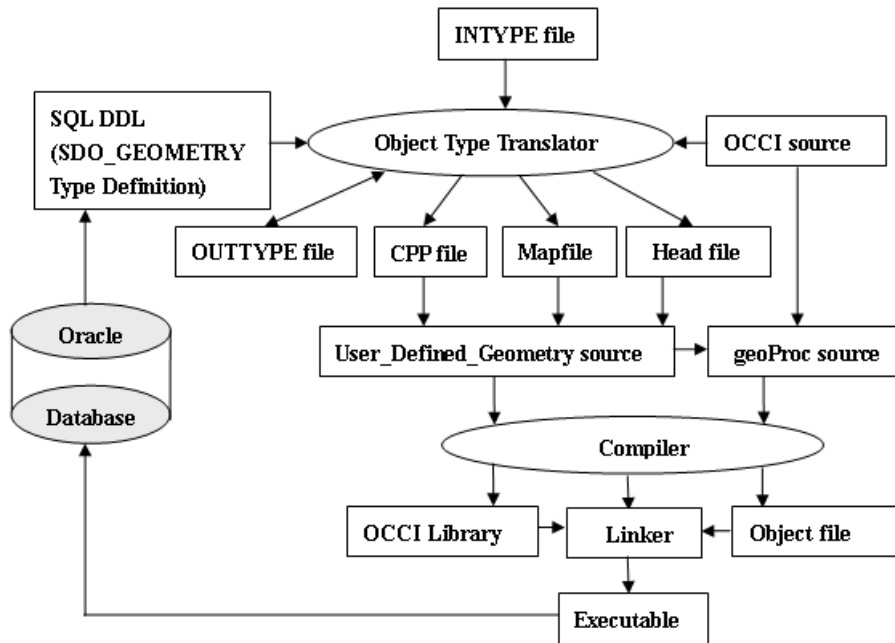
Fig. 1. The flow chart of accessing Oracle spatial database based on the OCCI

1) The implementation of client-side geometric object model User_Defined_Geometry

To make sure that the client-side can exchange spatial data with server-side SDO_GEOMETRY, it can establish a geometric object model User_Defined_Geometry in client-side which achieves the initialization operations of SDO_GEOMETRY object etc. The implementation steps of User_Defined_Geometry types are as follows:

a)   Write information file SDO_GEOMETRY.typ for transferring the SDO_GEOMETRY type.

b)   Transfer out class definition file and registration file of the object type SDO_GEOMETRY by object type translator OTT.

Ott userid=gis/gis_password   intype=SDO_GEOMETRY.typ

outtype=SDO_GEOMETRYout.typ code=cpp mapfile=SDO_GEOMETRYm.cpp
hfile=SDO_GEOMETRY.h cppfile=SDO_GEOMETRY.cpp attraccess=private

c)   Inherit the SDO_GEOMETRY:

Class User_Defined_Geometry: public SDO_GEOMETRY

{

Public:

User_Defined_Geometry（Corresponding parameters of the SDO_GEOMETRY type）;

~User_Defined_Geometry();

……   //Other user-defined extensions

 }

2) The implementation of the operation class GeoProc of client-side geometry object

This paper defines a class geoProc for manipulating the class User_Defined_Geometry based on the OCCI, the following is partial implementation:

Class geoProc

```
{
private:
Environment *env;
Connection *conn;
public:
//be used to create a connection to Oracle database
geoProc(string UserName,string Password,string db)
{
env=Environment::createEnvironment(Environment::OBJECT); //to create environment and register object
SDO_GEOMETRYm(env);
conn =env->createConnection (UserName,Password, db);
}
//Terminate connection and destroy environment variable geoProc::~geoProc()
{
env->terminateConnection (conn);
Environment::terminateEnvironment (env);
}
//Insert a User_Defined_Geometry object
Void InsertRow（string tableName,int id, User_Defined_Geometry * geo）
{
stmt= conn->createStatement ("insert into :x(objID, geoloc) values(:y, :z)");
Stmt=setString(1,tableName); //Bind parameters: x
Stmt=setInt(2,id); // Bind parameters: y
Stmt=setObject(3,geo); //Bind parameters: z
Stmt->execute();
Conn->commit();
}
//Get a User_Defined_Geometry object
User_Defined_Geometry * GetOneRow（string tableName ,int id）
{
User_Defined_Geometry * geo;
Stmt=conn->createStatement("select geoloc from :x where objID=:y");
…… //Bind parameters: x and y
resultSet * rs=stmt->executeQuery();
while(rs->next())
{
geo=(User_Defined_Geometry *)rs->getObject(1);
……    // Close resultset and release Stmt
}
}
……//Other operation functions, e.g., delete, update, etc.
}
```

3.2. Visualizing Spatial Data in Oracle Spatial by MapX

MapX can real-time access vector spatial data in Oracle spatial database through the bottom interface OCI[10]. Loading a layer is equivalent to downloading the corresponding spatial data from the server-side, and then add a new layer to the original map of the client-side .The key codes that MapX connects to Oracle Spatial are as follows:

```
//Define a layer object LayerInfo, and set every parameter
CMapXLayerInfo m_LayerInfo;
m_LayerInfo.CreateDispatch (m_LayerInfo.GetClsid ());
m_LayerInfo.SetType (miLayerInfoTypeServer);
m_LayerInfo.AddParameter ("ConnectString","SRVR=SDB;UID=gis;PWD=gis");
m_LayerInfo.AddParameter ("Name", "ZJ_JMM");
m_LayerInfo.AddParameter ("Toolkit", "ORAINET");
m_LayerInfo.AddParameter ("Query", "SELECT * FORM ZJ_JMM");
m_LayerInfo.AddParameter ("Visible", "1");
m_LayerInfo.AddParameter ("Cache", "ON");
m_LayerInfo.AddParameter ("Mbrsearch", "OFF");
//load and show the layer
m_ctrlMapX.GetLayers ().Add(m_LayerInfo,1);
```

3.3. Execute Spatial Operations Combining the Spatial Arithmetic Operators of Oracle Spatial

To compensate for the lack of spatial operation ability of MapX, the connection to Oracle via OCCI can be made, then call SQL statement that combines spatial operator of the Spatial to achieve a variety of complex spatial query and spatial analysis.

1) *Classification of spatial operations*
In general, spatial operators are performed through the SELECT statement，in order to facilitate description of the methods, according to the result of SELECT spatial operations,in this paper the spatial operations are divided into two categories:
   (1) Operations of executing result without geometric fields. For example, SELECT SDO_GEOM.SDO_LENGTH (geoloc,1) FROM SH_DTX WHERE ID=1, this statement is used to obtain the length of a Shanghai' subway line.
   (2) Operations of executing result with geometric fields. For example, SELECT A.geoloc FROM ZJ_JMQ A ,ZJ_JMQ B WHERE B.name='Linhai' AND SDO_FILTER(A.geoloc,B.geoloc)='TRUE'，executing this statement can get all the geometrics of residential areas which intersect possibly with Linhai City.

2) *The solution of spatial operations of executing result without geometric fields.*
Regarding this kind of spatial operation, direct call through the OCCI can be made and then the result of operation is handed over to the VC for display and related processing. For the case（1）,the following operation can be done:

```
Stmt=conn->createStatement("SELECT    SDO_GEOM.SDO_LENGTH   (geoloc,1)   FROM   SH_DTX
WHERE ID=1");
resultSet * rs=stmt->executeQuery();
while(rs->next())
{
   pLength=rs->getNumber(1); //obtain the measurement result
AfxMessageBox(str.Format("%f",pLength);//show the result
}
```

3) *The solution of spatial operations of executing result with geometric fields.*
Because this kind of spatial operation's result contains the spatial geometry type, VC++ cannot directly show the result, this paper carries out the visualization of the result with the aid of MapX. The key of this method is introducing a temporary layer for storing result of spatial operation, and the Query parameter of the corresponding CMapXLayerInfo object of this temporary layer is set to SQL statement with spatial operator

directly, then downloads from the Oracle Spatial and displays this layer, and highlights the result layer by the relevant CMapXSelection object. For the case (2), the query result is shown in Fig. 2.
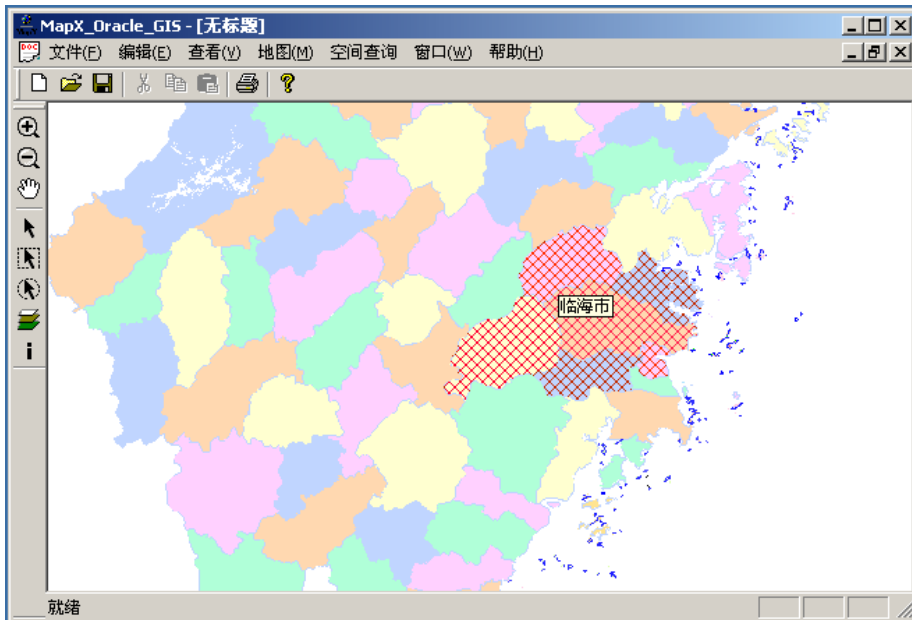

Fig. 2. The results' interface of spatial query

4. Conclusion

This paper implements client-side User_Defined_Geometry class and spatial data manipulative class geoProc on the basis of SDO_GEOMETRY and OCCI in VC ++6.0 environment, it makes reading, writing, inserting, deleting, and updating spatial data in Oracle Spatial become more convenient and simple. At the same time, this paper uses the powerful map functions and high visualization advantages of MapX component to realize various spatial operation and visualization process for spatial data in the Spatial, and calls spatial operators in Oracle Spatial to implement various high efficient and complex spatial operations by the OCCI, makes up for MapX's deficiencies in spatial operation functions. Finally, through the application example, it validates that developing GIS application system based on OCCI and MapX component is a very efficient and practical developing way.

References

[1] Oracle Corporation. Oracle 10g Application Developer's Guide-Object-Relational Features. June 2005.

[2] Oracle Corporation.Oracle Call Interface Programmer's Guide 10g Release 2. June 2005.

[3] Oracle Corporation.Pro*C/C++ Programmer's Guide 10g Release 2. June 2005.

[4] Oracle Corporation. Oracle C++ Call Interface Programmer's Guide 10g Release 2. June 2005.

[5] J. Quan. "Method and realization of accessing large object in database based on OCCI-Technology," Computer Systems & Applications, pp. 162-165,vol. 19,2010(In Chinese).

[6] R. C. Tan, L. Yao. "Design and Implementation of a construction survey information system based on MapX," Geomatics World, pp.66-69, 2007(In Chinese).

[7] K. Pu,B.B. He,X. W. Li. "Development of spatial database management system based on MapX and Oracle Spatial," Geospatial Information, pp.54-56,vol. 7,2009(In Chinese) .

[8] Oracle Corporation. Documentation for Oracle Spatial and Oracle Locator 10g. June 2005.

[9] X. H. Liang,Q. Tong. "Research on general process of accessing geographical data in Spatial by OCCI program," Microcomputer & Its Applications,pp.17-19,2004(In Chinese).

[10] Z. B. Ma, J. H. Wan. "The application research of Web GIS based on oracle spatial and mapX," Science of Surveying and Mapping, pp.106-108,vol.31,2006(In Chinese).