

License Configuration

Before you configuring license, please make sure that you has been obtained the license authorized by the Beijing Supermap Software Co., Ltd..

Getting official license

- Buying SuperMapGIS products to get license.
- Signing partners please contact SuperMap software partners sector, tel: + 86-10-59896655 transfer to 6167.
- Users and other partners please contact SuperMap software salesperson, tel: + 86-10-59896655 transfer to 6156.

Getting trial license

- Getting the trial license of SuperMap iMobile product through the online application, the trial license period is 3 months.

The address of applying for trial license online: <http://istore.supermap.com.cn/License/lic7C.aspx>

Getting cloud development online license

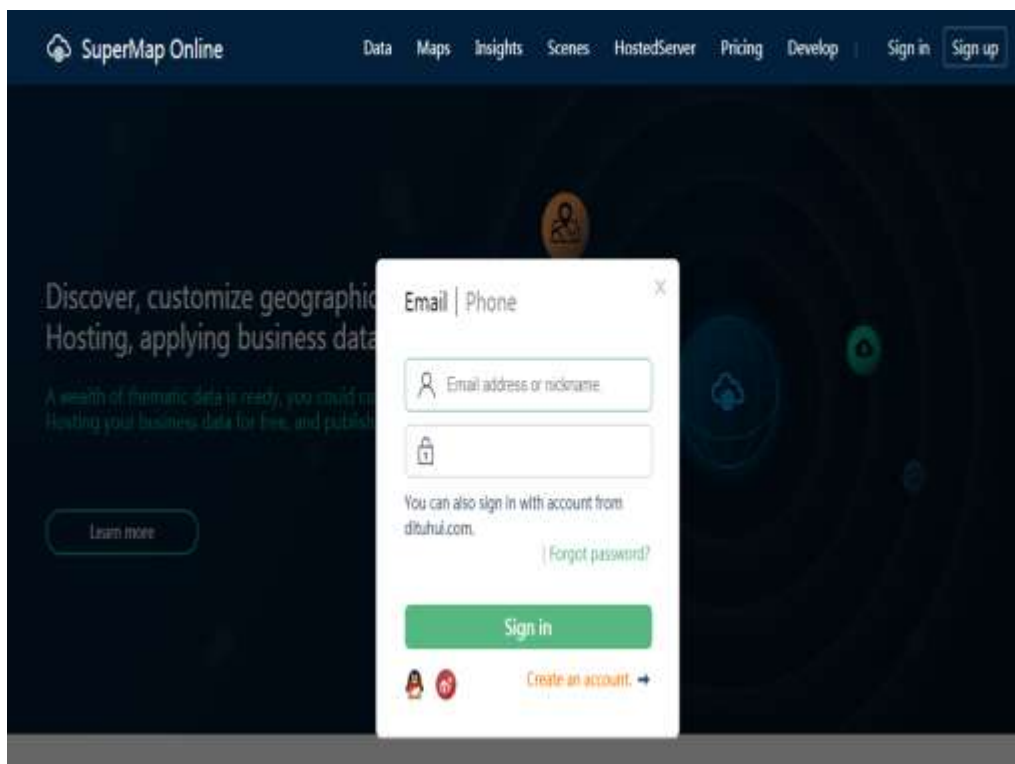
- Getting the cloud development online license of SuperMap iMobile product through the online application

The address of applying for trial license online: "<http://www.supermapol.com/>"

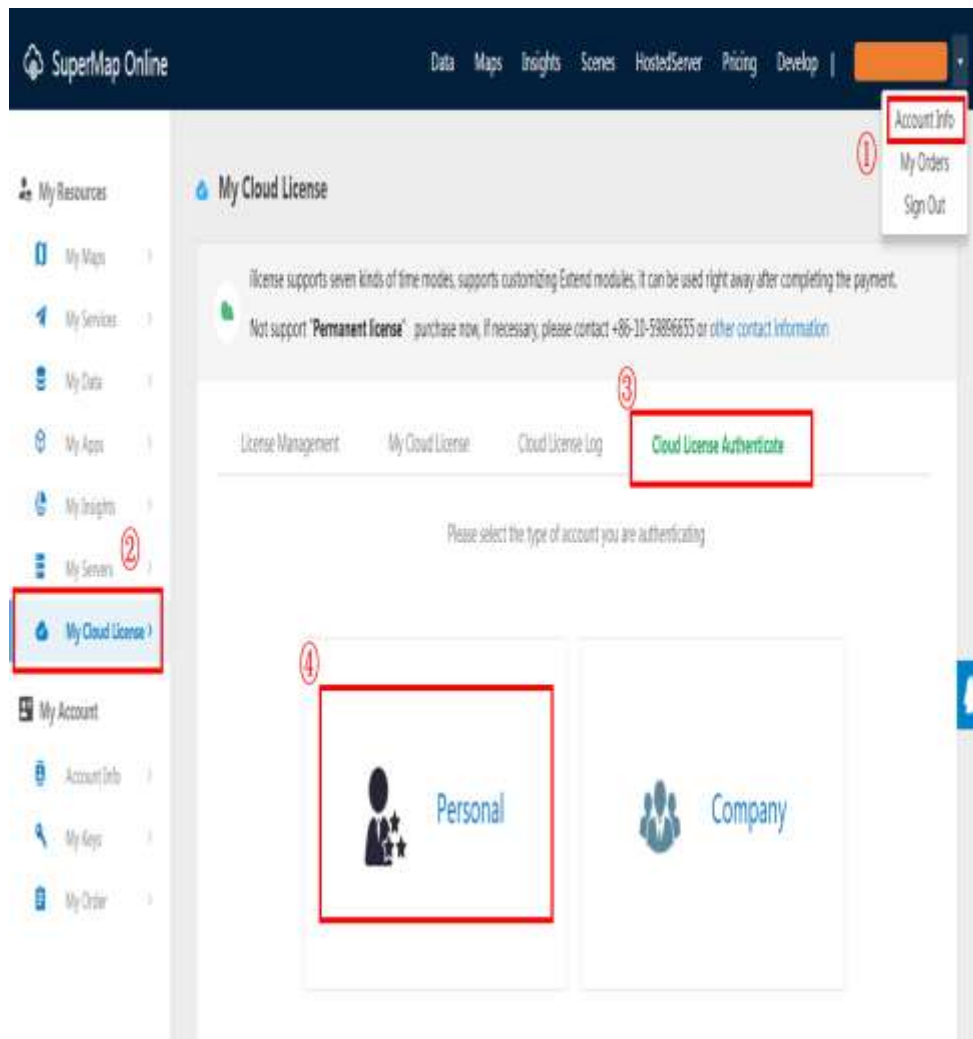
Note: 1. The effective use time of the cloud development online license is 1 day, if expired, users must apply again on the Internet. 2. The cloud development online license requests that the equipments must be real-time online.

Before you configuring license, please make sure that you have obtained the license authorized by the Beijing Supermap Software Co., Ltd..

- Obtaining the license of SuperMap iMobile products cloud online development and the detailed steps of configuration are as follows:
 1. Clicking [SuperMap Online](#) link, entering SuperMap Online home page, registering and logging according to the page prompt



2. After login success, click Account Info > My Cloud License > Cloud License Authenticate > Personal to go the authenticate page, and then click Bind now to bind you phone number.



- After completing to bind mobile phone number, returning to the "cloud license" page, clicking the "enterprise", filling in information according to page prompt, after completing to fill, clicking on the "submit" button to complete the enterprise information authentication.

License Management
My Cloud License
Cloud License Log
Cloud License Authenticate

* As the required information

* Username :

* Cell-phone number :

* Company name :
Be sure to keep in line with the name of the business license.

* Business license No :
Please check with the business license registration number or the unified social credit code number consistent!

* Scanning part of business license :
Please provide the original documents or color scanned photos, text / seal clearly identifiable.

- After completing the enterprise information authentication, namely configuring cloud development license in the application program codes, configuration codes are as follows:

```
private void cloudLicense(){

    String rootPath =
    android.os.Environment.getExternalStorageDirectory().getAbsolutePath();
    Environment.setLicensePath(rootPath + "/SuperMap/License");

    CloudLicenseManager cloudLicM = CloudLicenseManager.getInstance(this);
    String user = "Your online user name";
    String password = "Your online password";
    cloudLicM.setLoginCallback(new CloudLicenseManager.LicenseLoginCallback())
}
```

```
        @Override
        public void loginSuccess(LicenseStatus arg0) {
            // TODO Auto-generated method stub
            showMessage("Logging in cloudLicense successfully : " +
arg0.toString());
        }

        @Override
        public void loginFailed(String arg0) {
            // TODO Auto-generated method stub
            showMessage("Logging in cloudLicense fails: " + arg0);
        }
    });
    cloudLicM.login(user, password);
}
```

- The detailed steps for obtaining and configuring the trial license for SuperMap iMobile are as follows:
 1. Click the [link](#) for applying for trial license link online or enter the trial license application address in the browser address bar, jump to the trial license online application page, and enter information according to the page prompts.

Application information

Name: → **Your name**

Contact phone: → **Your phone number**

Email: → **Your real mail address**

The user name: → **Your username**

Company name: → **Name of your company**

Region: ☐ Ministry → **Please fill in your area carefully for timely and accurate service**

Product type: ☒ 9D ☐ 8C ☐ 7C ☐ 6R → **Product version**

Available products of the license: R:255 G:255 B:255

☒ SuperMap iDesktop

☐ SuperMap iServer ☐ SuperMap iPortal ☐ SuperMap iExpress ☐ SuperMap iManager

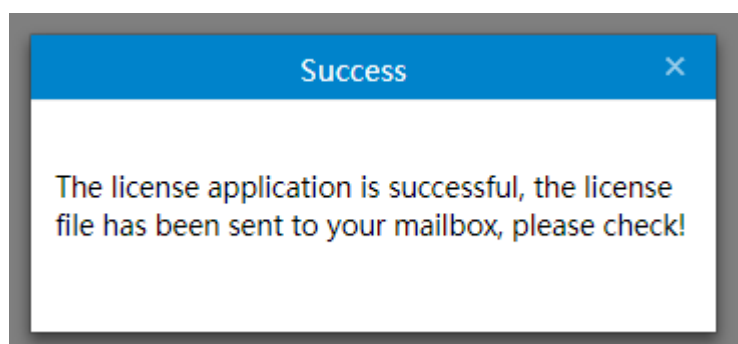
☐ SuperMap iObjects

☒ SuperMap iMobile → **Select SuperMap iMobile**

Way of encryption : ☒ Computer Name ☐ Operational report → **Sending an operational report is not necessary**

Computer name :

- After completing the application information, click the Submit button to submit your application. The successful application interface is as follows:



- Log in to the mailbox you filled out in the trial license application interface, and you will receive a message from online@supermap.com with the "Supermap license File" subject. The email contains the " SuperMap iMobile Trial.slm " file, which is the trial license for SuperMap iMobile, and this trial license applies to Android, iOS and is not bound to devices.

4. Get device default storage root directory String rootPath =
android.os.Environment.getExternalStorageDirectory().getAbsolutePath();
5. Place the trial license file "SuperMap iMobile Trial.slm" under the specified path of the device. In this example, the trial license is placed under *rootPath + "/SuperMap/license/"*.
6. Explicitly set the path to store the license file in the application initialization code through
Environment.setLicensePath(*rootPath + "/SuperMap/license/"*) (You can replace the path specified in the code as needed).
7. Till now, you have successfully configured the trial license for SuperMap iMobile.

Before you proceed with the license configuration, make sure you have obtained license from SuperMap Software Co., Ltd.

Product Official License Configuration

- **Offline License Configuration**

- A. Obtain the hardware identification code for the target device through the interface of Environment.getDeviceID (the hardware identification code for the device can be obtained directly by customizing the application).
- B. Obtain the official license activation file of the product through the sales channel with the obtained hardware identification code.
- C. Explicitly set the path to store the license file in the application initialization code through Environment.setLicensePath(*rootPath + "/SuperMap/license/"*) (You can replace the path specified in the code as needed).

Get device default storage root directory String rootPath =
android.os.Environment.getExternalStorageDirectory().getAbsolutePath();
- D. Store the official license activation file under the specified path.

- **Activate License Online**

- A. Initialization of the license manager;
- B. Set up a license activation callback;
- C. Set license type;
- D. Environment initialization
- E. Pass in user serial number and the list of modules that need to be activated to activate the device online;
- F. Restart the program after the device activation is successful.

Note: Sample code for configuring the online license by querying the passed functionality module is as follows:



```
ArrayList<Module> modules = new ArrayList<LicenseManager.Module>();
private void active() {

    //User serial number
    String userSerialID = "User serial number";

    RecycleLicenseManager manager =
RecycleLicenseManager.getInstance(getApplicationContext());

    //Set license activation callback
    manager.setActivateCallback(new RecycleLicenseCallback() {

        @Override
        public void upgradeFailed(String errorInfo) {
            System.out.println("upgradeFailed.");
        }

        @Override
        public void success(LicenseStatus newLicStatus) {
            System.out.println("success.");
        }

        @Override
        public void recycleLicenseFailed(String errorInfo) {
            System.out.println("recycle License Failed.");
        }

        @Override
        public void bindPhoneNumberFailed(String errorInfo) {
            System.out.println("bind Phone Number Failed.");
        }

        @Override
        public void activateFailed(String errorInfo) {
            System.out.println("activate Failed.");
        }
    }
}
```

```

        @Override
        public void queryResult(ArrayList<Module> arrModules) {
            modules = arrModules;
        }

        @Override
        public void otherErrors(String errorInfo) {
            System.out.println("other errors Failed.");
            System.out.println(errorInfo);
        }
    });

    //Set the license type (note: Set the license type first, and then initialize the environment)
    Environment.setLicenseType(LicenseType.UUID);
    Environment.initialization(getApplicationContext());
    //Query modules in the license
    manager.query(userSerialID);
    SystemClock.sleep(30 * 1000);
    //Activate device online and pass in user serial number and list of modules that need to
be activated
    manager.activateDevice(userSerialID, modules);
}

```

Note: Sample code for configuring the online license by adding modules directly is as follows:

```

private void active() {

    //User serial number
    String userSerialID = "User serial number";

    //Set modules that need to be activated
    final ArrayList<Module> modules = new ArrayList<LicenseManager.Module>();
    modules.add(Module.Core_Runtime);
    modules.add(Module.Navigation_Runtime);
    modules.add(Module.Realspace_Runtime);
}

```

```
RecycleLicenseManager manager =
RecycleLicenseManager.getInstance(getApplicationContext());

//Set license activation callback
manager.setActivateCallback(new RecycleLicenseCallback() {

    @Override
    public void upgradeFailed(String errorInfo) {
        System.out.println("upgradeFailed.");
    }

    @Override
    public void success(LicenseStatus newLicStatus) {
        System.out.println("success.");
    }

    @Override
    public void recycleLicenseFailed(String errorInfo) {
        System.out.println("recycle License Failed.");
    }

    @Override
    public void bindPhoneNumberFailed(String errorInfo) {
        System.out.println("bind Phone Number Failed.");
    }

    @Override
    public void activateFailed(String errorInfo) {
        System.out.println("activate Failed.");
    }

    @Override
    public void queryResult(ArrayList<Module> arrModules) {

    }

    @Override
    public void otherErrors(String errorInfo) {
```

```

        System.out.println("other errors Failed.");
        System.out.println(errorInfo);
    }
});

//Set liense type
Environment.setLicenseType(LicenseType.UUID);
Environment.initialization(getApplicationContext());
//Activate device online and pass in user serial number and list of modules that need to
be activated
manager.activateDevice(userSerialID, modules);
}

```

- **Backup License**

iMobile supports backing up licenses by binding the phone number. After the license has been successfully backed up, once the device is lost or scrapped, the license can be returned through the return interface on other devices, and the returned license can be redistributed.

Note:

1. A mobile number can only back up one license
2. The backup operation can be done after the license activation

Note: The sample code for license backup is as follows:

```

private void bind() {

    RecycleLicenseManager manager =
RecycleLicenseManager.getInstance(getApplicationContext());
    //Pass in the phone number and bind the license
    manager.bindPhoneNumber(mPhoneNumber);
}

```

- **Return License**

iMobile supports returning license. Licenses that have been backed up by a cell phone number can be returned via passing in the phone number. If a license has not been backed up, it can be returned if the parameter is null.

Note: Return operation is allowed after license activation

Note: The sample code for returning license is as follows:

```
private void recycle() {  
  
    RecycleLicenseManager manager =  
    RecycleLicenseManager.getInstance(getApplicationContext());  
    //Pass in the phone number and return the license  
    manager.recycleLicense(mPhoneNumber);  
}
```

- **Upgrade License**

Products below the 9D version can be upgraded.

Note: The sample code for upgrading license is as follows:

```
private void upgrade() {  
  
    RecycleLicenseManager manager =  
    RecycleLicenseManager.getInstance(getApplicationContext());  
    //Pass in original contract number and new contract number to upgrade license  
    manager.upgrade(mUserSerialID, mUserSerialIDNew);  
}
```