Cordova—iOS教程

一、Cordova说明：

Cordova提供了一组设备相关的API，通过这组API，[移动应用](http://cpro.baidu.com/cpro/ui/uijs.php?adclass=0&app_id=0&c=news&cf=1001&ch=0&di=128&fv=18&is_app=0&jk=eada76038ca7df3&k=%D2%C6%B6%AF%D3%A6%D3%C3&k0=%D2%C6%B6%AF%D3%A6%D3%C3&kdi0=0&luki=6&n=10&p=baidu&q=85048100_cpr&rb=0&rs=1&seller_id=1&sid=f37dca3860a7ad0e&ssp2=1&stid=0&t=tpclicked3_hc&td=1797308&tu=u1797308&u=http%3A%2F%2Fwww%2Ecodes51%2Ecom%2Farticle%2Fdetail%5F125775%2Ehtml&urlid=0" \t "_blank)能够以JavaScript访问原生的设备功能，如[摄像头](http://cpro.baidu.com/cpro/ui/uijs.php?adclass=0&app_id=0&c=news&cf=1001&ch=0&di=128&fv=18&is_app=0&jk=eada76038ca7df3&k=%C9%E3%CF%F1%CD%B7&k0=%C9%E3%CF%F1%CD%B7&kdi0=0&luki=1&n=10&p=baidu&q=85048100_cpr&rb=0&rs=1&seller_id=1&sid=f37dca3860a7ad0e&ssp2=1&stid=0&t=tpclicked3_hc&td=1797308&tu=u1797308&u=http%3A%2F%2Fwww%2Ecodes51%2Ecom%2Farticle%2Fdetail%5F125775%2Ehtml&urlid=0" \t "_blank)、[麦克风](http://cpro.baidu.com/cpro/ui/uijs.php?adclass=0&app_id=0&c=news&cf=1001&ch=0&di=128&fv=18&is_app=0&jk=eada76038ca7df3&k=%C2%F3%BF%CB%B7%E7&k0=%C2%F3%BF%CB%B7%E7&kdi0=0&luki=2&n=10&p=baidu&q=85048100_cpr&rb=0&rs=1&seller_id=1&sid=f37dca3860a7ad0e&ssp2=1&stid=0&t=tpclicked3_hc&td=1797308&tu=u1797308&u=http%3A%2F%2Fwww%2Ecodes51%2Ecom%2Farticle%2Fdetail%5F125775%2Ehtml&urlid=0" \t "_blank)等。 Cordova还提供了一组统一的JavaScript类库，以及为这些类库所用的设备相关的原生后台代码。 Cordova支持如下移动操作系统：iOS, Android,ubuntu phone os, Blackberry, Windows Phone, Palm WebOS, Bada 和 Symbian。

Cordova是贡献给**[Apache](http://www.codes51.com/article/search_Apache/" \t "_blank)**后的[开源](http://cpro.baidu.com/cpro/ui/uijs.php?adclass=0&app_id=0&c=news&cf=1001&ch=0&di=128&fv=18&is_app=0&jk=eada76038ca7df3&k=%BF%AA%D4%B4&k0=%BF%AA%D4%B4&kdi0=0&luki=5&n=10&p=baidu&q=85048100_cpr&rb=0&rs=1&seller_id=1&sid=f37dca3860a7ad0e&ssp2=1&stid=0&t=tpclicked3_hc&td=1797308&tu=u1797308&u=http%3A%2F%2Fwww%2Ecodes51%2Ecom%2Farticle%2Fdetail%5F125775%2Ehtml&urlid=0" \t "_blank)项目，是从PhoneGap中抽出的核心代码，是驱动PhoneGap的核心引擎。你可以把他想象成类似于Webkit和Google Chrome的关系。

二、环境配置，新建项目

1. 如果你的terminal不能运行 npm，那先得安装 Node.js

Node文件夹下的node-v0.12.7.pkg

下载链接：<http://nodejs.org/download/>

2. 在命令行terminal利用 npm安装 Cordova

sudo npm install -g cordova

3. 开始建立第一个CordovaIOSPrj的应用

3.1 创建项目

cordova create CordovaIOSPrj com.example.CordovaIOSPrj IOSPrj -d

-d 是为了在过程中能输出信息。

之后，在你运行上面这条命令的路径下，就会建立一个"CordovaIOSPrj"的目录。在CordovaIOSPrj目录下有一个 "www"的目录，将是你应用的hompage的目录。

3.2 cd到CordovaIOSPrj目录，到执行命令，为CordovaIOSPrj添加iOS的平台支持

cordova platform add ios

执行成功后，在CordovaIOSPrj/platforms/下面就会多了一个ios的目录了。

3.3 运行下面命令build项目

cordova build

一堆信息过后，最后出现"BUILD SUCCEEDED"

之后，就可以，在xCode中打开该项目。选择"platforms/ios/IOSPrj.xcodeproj"文件打开。

运行成功：



三、添加plugin

添加设备API:

cordova plugin add cordova-plugin-device

网络连接：

cordova plugin add cordova-plugin-network-information

电池：

cordova plugin add cordova-plugin-battery-status

加速度计：

cordova plugin add cordova-plugin-device-motion

地理定位：

cordova plugin add cordova-plugin-geolocation

相机：

cordova plugin add cordova-plugin-camera

媒体重播和捕获：

cordova plugin add cordova-plugin-media-capture

cordova plugin add cordova-plugin-media

联系人：

cordova plugin add cordova-plugin-contacts

其他参考Apache Cordova Documentation

四、加入html，CSS，JS等文件

运行效果图：



五、常用Native API的使用

**5.1 Accelerometer(加速计传感器)**

html代码：

<!-- Accelerometer

$ cordova plugin add org.apache.cordova.device-motion

-->

<div data-role="page" id="accelerometer">

<div data-role="header">

<a data-role="button" data-rel="back" data-direction="reverse" data-icon="arrow-l">Back</a>

<h1>Accelerometer</h1>

</div>

<div data-role="content">

<a href="#" data-role="button" id="startWatch">Start Watching</a><br>

<a href="#" data-role="button" id="stopWatch">Stop Watching</a><br>

<div id="accvals">Waiting for accelerometer...</div>

<br><br>

<a href="#" data-role="button" id="startWatchOrientation">Start Watch Orientation</a><br>

<a href="#" data-role="button" id="stopWatchOrientation">Stop Watch Orientation</a><br>

<div id="orivals">Waiting for orientation...</div>

</div>

<script type="text/javascript">

var watchID = null;

document.addEventListener('deviceready', onDeviceReady, false);

function onDeviceReady() {

$("#startWatch").on("click", startWatch);

$("#stopWatch").on("click", stopWatch);

$("#startWatchOrientation").on("click", startWatchOrientation);

$("#stopWatchOrientation").on("click", stopWatchOrientation);

}

function startWatch() {

alert("startWatch");

var options = { frequency: 3000 };

watchID = navigator.accelerometer.watchAcceleration(onAccelSuccess, onAccelError, options);

}

function stopWatch() {

alert("stopWatch");

if (watchID) {

navigator.accelerometer.clearWatch(watchID);

watchID = null;

}

}

function onAccelSuccess(acceleration) {

var element = document.getElementById('accvals');

element.innerHTML = '<strong>Accel X:</strong> ' + acceleration.x.toFixed(1) \* -1 + '<br />' +

'<strong>Accel Y:</strong> ' + acceleration.y.toFixed(1) + '<br />' +

'<strong>Accel Z:</strong> ' + acceleration.z.toFixed(1) + '<br />' +

'<strong>Timestamp:</strong> ' + acceleration.timestamp + '<br />';

}

function onAccelError() {

alert('Could not Retrieve Accelerometer Data!');

}

function deviceOrientationEvent(eventData) {

var alpha = Math.round(eventData.alpha);

var beta = Math.round(eventData.beta);

var gamma = Math.round(eventData.gamma);

var element = document.getElementById('orivals');

element.innerHTML = ("alpha = " + alpha + " beta = " + beta + " gamma = " + gamma);

}

function startWatchOrientation() {

alert("startWatchOrientation");

window.addEventListener('deviceorientation', deviceOrientationEvent);

}

function stopWatchOrientation() {

alert("stopWatchOrientation");

window.removeEventListener('deviceorientation', deviceOrientationEvent);

}

</script>

</div>

**5.2 Camera(摄像头)**

html代码：

<!-- Camera

$ cordova plugin add org.apache.cordova.camera

-->

<div data-role="page" id="camera">

<div data-role="header">

<a data-role="button" data-rel="back" data-direction="reverse" data-icon="arrow-l">Back</a>

<h1>Camera</h1>

</div>

<div data-role="content">

<a href="#" data-role="button" id="capturePhoto">Capture Photo</a><br>

<img style="display:none;" id="smallImage" src="" /><p id="smTitle"></p>

<a href="#" data-role="button" id="browsePhoto">Browse Photo Album</a><br>

<img style="display:none;" id="largeImage" src="" /><p id="lgTitle"></p>

</div>

<script type="text/javascript">

var pictureSource;

var destinationType; //

document.addEventListener('deviceready', onDeviceReady, false);

function onDeviceReady() {

pictureSource = navigator.camera.PictureSourceType;

destinationType = navigator.camera.DestinationType;

$("#capturePhoto").on("click", capturePhoto);

$("#browsePhoto").on("click", browsePhoto);

}

function capturePhoto() {

alert("capturePhoto");

if (!navigator.camera) {

alert("Camera API not supported", "Error");

return;

}

navigator.camera.getPicture(onPhotoDataSuccess, onFail,

{ quality: 50,

allowEdit: true,

destinationType: destinationType.DATA\_URL });

}

function browsePhoto() {

alert("browsePhoto");

navigator.camera.getPicture(onPhotoURISuccess, onFail,

{ quality: 50,

destinationType: destinationType.FILE\_URI,

sourceType: pictureSource.PHOTOLIBRARY });

}

//sourceType 0:Photo Library, 1=Camera, 2=Saved Album

//encodingType 0=JPG 1=PNG

function onFail(message) {

alert('Response: ' + message);

}

function onPhotoDataSuccess(imageData) {

var smallImage = document.getElementById('smallImage');

smallImage.style.display = 'block';

smallImage.src = "data:image/jpeg;base64," + imageData;

}

function onPhotoURISuccess(imageURI) {

var largeImage = document.getElementById('largeImage');

largeImage.style.display = 'block';

largeImage.src = imageURI;

}

</script>

</div>

**5.3 Device(设备信息)**

html代码：

<!-- Device

$ cordova plugin add org.apache.cordova.device

-->

<div data-role="page" id="device">

<div data-role="header">

<a data-role="button" data-rel="back" data-direction="reverse" data-icon="arrow-l">Back</a>

<h1>Device</h1>

</div>

<div data-role="content">

<a href="#" data-role="button" id="getDeviceProperties">Get Device Properties</a><br>

<div id="deviceProperties">Loading device properties...</div>

</div>

<script type="text/javascript">

document.addEventListener("deviceready", onDeviceReady, false);

function onDeviceReady() {

$("#getDeviceProperties").on("click", getDeviceProperties);

}

function getDeviceProperties() {

alert("getDeviceProperties");

var element = document.getElementById('deviceProperties');

element.innerHTML = 'Device Model (Android: product name): ' + device.model + '<br />' +

'Cordova version: ' + device.cordova + '<br />' +

'Operating system: ' + device.platform + '<br />' +

'Device UUID: ' + device.uuid + '<br />' +

'Operating system version: ' + device.version + '<br />';

}

</script>

</div>

**5.4 Connection(网络连接状态)**

**html代码：**

<!-- Connection

$ cordova plugin add org.apache.cordova.network-information

-->

<div data-role="page" id="connection">

<div data-role="header">

<a data-role="button" data-rel="back" data-direction="reverse" data-icon="arrow-l">Back</a>

<h1>Connection</h1>

</div>

<div data-role="content">

<a href="#" data-role="button" id="checkConnection">Check Connection</a><br>

<div id="connectiontype">Waiting for Connection type...</div>

</div>

<script type="text/javascript">

document.addEventListener("deviceready", onDeviceReady, false);

function onDeviceReady() {

$("#checkConnection").on("click", checkConnection);

}

function checkConnection() {

alert("checkConnection");

var networkState = navigator.connection.type;

var states = {};

states[Connection.UNKNOWN] = 'Unknown connection';

states[Connection.ETHERNET] = 'Ethernet connection';

states[Connection.WIFI] = 'WiFi connection';

states[Connection.CELL\_2G] = 'Cell 2G connection';

states[Connection.CELL\_3G] = 'Cell 3G connection';

states[Connection.CELL\_4G] = 'Cell 4G connection';

states[Connection.CELL] = 'Cell generic connection';

states[Connection.NONE] = 'No network connection';

var element = document.getElementById('connectiontype');

element.innerHTML = 'Connection type: ' + states[networkState];

}

</script>

</div>

**5.5 Geolocation(GPS地理位置服务)**

html代码：

<!-- Geolocation

$ cordova plugin add org.apache.cordova.geolocation

-->

<div data-role="page" id="geolocation">

<div data-role="header">

<a data-role="button" data-rel="back" data-direction="reverse" data-icon="arrow-l">Back</a>

<h1>Geolocation</h1>

</div>

<div data-role="content">

<a href="#" data-role="button" id="startGeolocationg">Start Geolocationg</a><br>

<a href="#" data-role="button" id="stopGeolocationg">Stop Geolocation</a><br>

<br><br>

<a href="#" data-role="button" id="getCurrentPosition">Get Current Position </a><br>

<div id="geovals">Waiting for geolocation...</div>

</div>

<script type="text/javascript">

var watchGeoID = null;

document.addEventListener("deviceready", onDeviceReady, false);

function onDeviceReady() {

$("#startGeolocationg").on("click", startGeolocationg);

$("#stopGeolocationg").on("click", stopGeolocationg);

$("#getCurrentPosition").on("click", getCurrentPosition);

}

function startGeolocationg() {

alert('startGeolocationg');

var element = document.getElementById('geovals');

element.innerHTML = 'Finding geolocation every 30 seconds...'

var options = { enableHighAccuracy: true, timeout: 30000 };

watchGeoID = navigator.geolocation.watchPosition(onGeoSuccess, onGeoError, options);

}

function onGeoSuccess(position) {

var element = document.getElementById('geovals');

element.innerHTML =

'<strong>Latitude:</strong> ' + position.coords.latitude + '<br />' +

'<strong>Longitude: </strong> ' + position.coords.longitude + ' <br />' +

'<strong>Altitude</strong> (in meters): ' + position.coords.altitude + ' <br />' +

'<strong>Accuracy</strong> (in meters): ' + position.coords.accuracy + ' <br />' +

'<strong>Altitude Accuracy:</strong> ' + position.coords.altitudeAccuracy + ' <br />' +

'<strong>Heading</strong> (direction of travel): ' + position.coords.heading + ' <br />' +

'<strong>Speed</strong> (meters per second): ' + position.coords.speed + ' <br />' +

'<strong>Timestamp:</strong> ' + position.timestamp + ' <br />';

}

function onGeoError(error) {

var element = document.getElementById('geovals');

element.innerHTML =+ '<br>' + error.code + error.message;

}

function stopGeolocationg() {

alert('stopGeolocationg');

var element = document.getElementById('geovals');

element.innerHTML = '<span style="color:red">Geolocation turned off.</span>';

if (watchGeoID) {

navigator.geolocation.clearWatch(watchGeoID);

watchGeoID = null;

}

}

function getCurrentPosition() {

alert('getCurrentPosition');

navigator.geolocation.getCurrentPosition(onPositionSuccess, onPositionError);

}

function onPositionSuccess(position) {

var element = document.getElementById('geovals');

element.innerHTML =+ ('Latitude: ' + position.coords.latitude + '\n' +

'Longitude: ' + position.coords.longitude + '\n');

};

function onPositionError(error) {

var element = document.getElementById('geovals');

element.innerHTML =+('Error getting GPS Data');

}

</script>

</div>

**5.6 File(文件系统处理 )**

html代码：

<!-- File

$ cordova plugin add org.apache.cordova.file

$ cordova plugin add org.apache.cordova.file-transfer

-->

<div data-role="page" id="file">

<div data-role="header">

<a data-role="button" data-rel="back" data-direction="reverse" data-icon="arrow-l">Back</a>

<h1>File</h1>

</div>

<div data-role="content">

<input type="text" id="userInput" name="userInput" value="I'm rensanning."><br>

<a href="#" data-role="button" id="writeToFile">Write To File</a><br>

<a href="#" data-role="button" id="readFile">Read File</a><br>

<p id="data1"></p><p id="data2"></p><br>

<a href="#" data-role="button" id="deleteFile">Delete File</a><br>

</div>

<script type="text/javascript">

document.addEventListener("deviceready", onDeviceReady, false);

function onDeviceReady() {

$("#writeToFile").on("click", writeToFile);

$("#readFile").on("click", readFile);

$("#deleteFile").on("click", deleteFile);

}

function writeToFile() {

window.requestFileSystem(LocalFileSystem.PERSISTENT, 0, gotFSForWrite, fail);

}

function gotFSForWrite(fileSystem) {

fileSystem.root.getFile("CordovaSample.txt", {create: true, exclusive: false}, gotWriteFileEntry, fail);

}

function gotWriteFileEntry(fileEntry) {

fileEntry.createWriter(gotFileWriter, fail);

}

function gotFileWriter(writer) {

var userText = $('#userInput').val();

writer.seek(writer.length);

writer.write('\n\n' + userText);

writer.onwriteend = function(evt){

alert("You wrote ' " + userText + " ' at the end of the file.");

}

$('#userInput').val("");

}

function readFile() {

window.requestFileSystem(LocalFileSystem.PERSISTENT, 0, gotFSForRead, fail);

}

function gotFSForRead(fileSystem) {

fileSystem.root.getFile("CordovaSample.txt", null, gotReadFileEntry, fail);

}

function gotReadFileEntry(fileEntry) {

fileEntry.file(gotFileRead, fail);

}

function gotFileRead(file){

readDataUrl(file);

readAsText(file);

}

function readDataUrl(file) {

var reader = new FileReader();

reader.onloadend = function(evt) {

var element = document.getElementById('data1');

element.innerHTML = '<strong>Read as data URL:</strong> <br><pre>' + evt.target.result + '</pre>';

};

reader.readAsDataURL(file);

}

function readAsText(file) {

var reader = new FileReader();

reader.onloadend = function(evt) {

var element = document.getElementById('data2');

element.innerHTML = '<strong>Read as data text:</strong> <br><pre>' + evt.target.result + '</pre>';

};

reader.readAsText(file);

}

function deleteFile() {

window.requestFileSystem(LocalFileSystem.PERSISTENT, 0, gotFSForRemove, fail);

}

function gotFSForRemove(fileSystem) {

fileSystem.root.getFile("CordovaSample.txt", {create: false, exclusive: false}, gotRemoveFileEntry, fail);

}

function gotRemoveFileEntry(fileEntry){

fileEntry.remove(

function(entry) {

alert("Removal succeeded");

},

function(error) {

alert("Error removing file: " + error.code);

});

}

function fail(error) {

alert(error.code);

}

</script>

</div>

**5.7 Database(客户端数据库)**

html代码：

<!-- Database -->

<div data-role="page" id="database">

<div data-role="header">

<a data-role="button" data-rel="back" data-direction="reverse" data-icon="arrow-l">Back</a>

<h1>Database</h1>

</div>

<div data-role="content">

<label for="id">ID:</label>

<input type="text" id="id" name="id" value="12345"><br>

<label for="data">Data:</label>

<input type="text" id="data" name="data" value="Data Value"><br>

<a href="#" data-role="button" id="saveToDatabase">Save To Database</a><br>

<a href="#" data-role="button" id="getFromDatabase">Get From Database</a><br>

</div>

<script type="text/javascript">

var db;

document.addEventListener("deviceready", onDeviceReady, false);

function onDeviceReady() {

$("#saveToDatabase").on("click", saveToDatabase);

$("#getFromDatabase").on("click", getFromDatabase);

db = window.openDatabase("MyDatabase", "1.0", "Cordova Sample", 200000);

db.transaction(function(tx) {

tx.executeSql('DROP TABLE IF EXISTS MyTable');

tx.executeSql('CREATE TABLE IF NOT EXISTS MyTable (id unique, data)');

},

function(err) {

alert("Error processing SQL: " + err.code);

});

}

function saveToDatabase() {

alert('saveToDatabase');

db.transaction(function(tx) {

tx.executeSql("INSERT INTO MyTable (id, data) VALUES (?, ?)",

[$('#id').val(), $('#data').val()],

function(tx, rs) {

alert("Your SQLite query was successful!");

},

function(tx, e) {

alert("SQLite Error: " + e.message);

});

});

}

function getFromDatabase() {

alert('getFromDatabase');

db.transaction(function(tx) {

tx.executeSql("SELECT id,data FROM MyTable ORDER BY id",

[],

function (tx, rs) {

for (var i = 0; i < rs.rows.length; i++) {

alert(rs.rows.item(i).id + "=" + rs.rows.item(i).data);

}

},

function(tx, e) {

alert("SQLite Error: " + e.message);

});

});

}

</script>

</div>