
MORE Storage Space Controller SDK

Version 0.16.08.15

2016.08.15

Table of Contents

MORE Storage Space Controller SDK.....	1
<i>Version 0.16.03.04</i>	1
1. Introduction.....	3
1.1. Overview	3
2. Service.....	4
2.1. Storage Space Controller SDK	4
2.2. Use SDK Library	4
2.3. StorageSpace Function.....	7
3. Parameter Definition	7
3.1. OnCallbackResult Function Parameter.....	7
3.2. Handler 與 Callback listener 參數對應.....	8

1. Introduction

1.1. Overview

MORE Storage Space Controller SDK 主要提供 Smart Mobile Device 更快速與簡潔的開發模組，APP 開發人員只要將 MORE Storage Space Controller SDK 加入到自己的 APP 專案裡，透過 API 的呼叫即可知道智慧型裝置目前儲存空間剩餘多少空間，目前提供 Android 版本 SDK Library。

2. Service

2.1. Storage Space Controller SDK

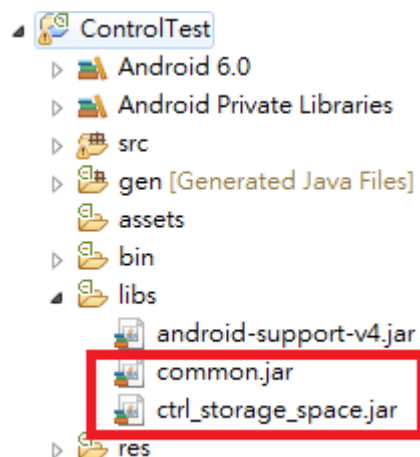
Storage Space Controller SDK 主要提供給開發者一個能主動通知開發者目前智慧型裝置剩餘可用的儲存空間。

2.2. Use SDK Library

在使用 Storage Space Controller SDK 前，開發者可以設定的儲存空間區間差內要回報，此模組會透過 Handler 或 Callback Listener 主動通知開發者目前儲存空間剩餘量。

使用步驟：

1. 先將以下 jar 檔加入到 Android 開發專案的 libs 資料夾



2. 匯入以下 library

```
import android.os.Bundle;
import android.os.Handler;
import android.os.Message;
import android.util.Log;
import sdk.ideas.common.CtrlType;
import sdk.ideas.common.OnCallbackResult;
import sdk.ideas.common.ResponseCode;
import sdk.ideas.ctrl.space.StorageSpaceHandler;
import java.util.HashMap;
```

3. 設定Storage Space Handler

```
StorageSpaceHandler mStorageSpaceHandler = new StorageSpaceHandler(this);
```

4. 本SDK 提供Handler 和Callback Listener方法來去知道程式是否有成功執行，開發者可以以自己喜好去做設定，以下是Handler 方法範例
- 1) 先宣告Handler

```
private Handler theHandler = new Handler()
{
    @Override
    public void handleMessage(Message msg)
    {
        switch (msg.what)
        {
            case CtrlType.MSG_RESPONSE_STORAGE_SPACE_HANDLER:
                Log.d("handler response: ", "Result: " + String.valueOf(msg.arg1) + " From: "
                    + String.valueOf(msg.arg2) + " Message: " +
                    ((HashMap<String,String>)msg.obj) );
                break;
            default:
                break;
        }
    }
};
```

2) 設定此 handler

```
mStorageSpaceHandler.setHandler(theHandler);
```

5. Callback Listener 方式

```
mStorageSpaceHandler.setOnCallbackResultListener(new OnCallbackResult()
{
    @Override
    public void onCallbackResult(int result, int what, int from, HashMap<String,
String> message)
    {
        Log.d("Listener response: ", "Result: " + String.valueOf(result) + " What:
" + String.valueOf(what)
            + " From: " + String.valueOf(from) + " " + message);
    }

});
```

6. 設定時間區間跟儲存空間差，預設是 1 分鐘，50MB，當跟舊紀錄相差
50MB 即會回報。

```
mStorageSpaceHandler.setCheckTime(60000);
mStorageSpaceHandler.setDiffStorageSpace(50);
```

7. 啟動 Storage Space handler

```
mStorageSpaceHandler.startListenAction();
```

8. 關閉 Storage Space handler

```
mStorageSpaceHandler.stopListenAction();
```

2.3. StorageSpace Function

StorageSpace Controller API	
Function	Description
StorageSpaceHandler (Context mContext)	建構元
void setCheckTime (long millisecondTime)	設定多少時間去檢查儲存空間差，預設是 1 分鐘
void setDiffStorageSpace (float diffStorageSpaceMB)	設定多少空間差要回報，預設是跟舊儲存空間差 50MB 即回報
void startListenAction()	開始監看儲存空間服務
void stopListenAction()	停止監看儲存空間服務
void setHandler(Handler handler)	設定 Handler，參數對應可參考 3.2
void setOnCallbackResultListener(OnCallbackResult listener)	設定 callback listener，其參數參考 3.1

3. Parameter Definition

3.1. OnCallbackResult Function Parameter

result :

Value	Description
1	SUCCESS
0	UNKNOWN ERROR
-19	EXTERNAL MEMORY UNAVAILABLE ERROR

what :

Value	Description
1036	STORAGE SPACE CONTROLLER HANDLER

from :

Value	Description
0	EXTERNAL MEMORY
1	REMOVABLE EXTERNAL MEMORY

message :

key	Description
“message”	敘述執行成功或錯誤訊息
“availablememory”	尚可用儲存空間大小(MB)
“totalmemory”	全部儲存空間(MB)

3.2. Handler 與 Callback listener 參數對應

Handler	OnCallbackResult
msg.what	int what
msg.arg1	int result
msg.arg2	int from
msg.obj	HashMap<String, String> message