

# Windows下php擴展開發c++動態庫

分類： 編程 (/list/編程/1/) 時間:2016-11-08

[摘要：PHP擴大開辟，從整懂得到開端完成一個小項目，經由三天的細致研討，現整頓以下 1、需供先容 PHP擴大開辟，挪用本身之前的c++靜態庫，完樂成能 2、項目之前 體系：windo]

PHP擴展開發，從零了解到初步完成一個小項目，經過三天的仔細研究，現整理如下

## 一、需求介紹

PHP擴展開發，調用自己之前的c++動態庫，完成功能

## 二、項目之前

系統：Windows xp

開發工具：vs 2008

web環境：apache2.4 PHP5.3.29-VC9-ts-x86 aphach和PHP 環境之前已經搭建完成

PHP源碼：去官網<http://www.php.net/downloads.php> 下載穩定版本的php源碼包（因為要編譯擴展庫，必須要php的源碼才能編譯），將源碼解壓到如：d:\php\_src目錄下。本示例用的是PHP5.3.29。下載二進制包（如果已經安裝了php環境，就可以不用下載），這裏主要用到php二進制包中的php5ts.lib，該文件位於php的dev目錄。本示例使用

相關文章



配置源碼：將源碼中php\_src/win32/build/config.w32.h.in文件拷貝一份到php\_src/main/下，並重命名為：config.w32.h。

PHP二進制包

PHP源碼包

## 三、創建項目

1、創建一個空的win32項目（注意：是Win32的 dll 項目工程，）。

2、配置工程屬性：

（1）添加附加包含目錄：在C/C++的選項中，添加附加包含目錄。包含php源碼中的幾個目錄。

如：D:\php\_src;D:\php\_src\main;D:\php\_src\Zend;D:\php\_src\TSRM;D:\php\_src\win32;

（2）添加預處理器：ZEND\_DEBUG=0;ZTS=1;ZEND\_WIN32;PHP\_WIN32;

ADVERTISEMENT



- 環信即時通訊在Android平臺 (/articles/1478578237.htm)
- Android內存和SD卡的數據 (/articles/1478578244.htm)
- Android系統的音量控制 (/articles/1478578506.htm)
- [置頂] Android服務——Se (/articles/1478578511.htm)
- android-eclipse編寫aidl的 aidl.exe error while creati directories (/articles/1478578519.htm)
- Android-Activity中的四個狀 (/articles/1478578522.htm)
- Android-BaseAdapter中重的心得以及發現convertView制 (/articles/1478578525.htm)
- Android-Parcelable接口的 (/articles/1478578528.htm)
- Android WebView JS交互; WebView中的圖片點擊事件 (/articles/1478578531.htm)
- C和C++的讀入方式對比 (/articles/1478578535.htm)

ad



( 3 ) 添加附加庫：php5ts.lib ( 該庫位於php二進制文間包中的dev目錄 )

#### 四、編寫源碼示例

1、添加源文件如：Main.cpp和源文件的頭文件Main.h。其中文件的內容主要參考了在linux下編寫php擴展庫，自動生成的文件的內容 ( cygwin

可以幫助實現搭建好擴展的骨架，我沒試驗過，不過沒有cygwin也沒關係，直接拷貝下面的代碼 )

Main.h文件內容：

```

#ifndef PHP_TEST_MAIN_H
#define PHP_TEST_MAIN_H

extern zend_module_entry PHPTest_module_entry;          // PHPTest 是該示例的工程名字， PHPTes
t_module_entry是php擴展庫的入口聲明
#define phpext_PHPTest_ptr &PHPTest_module_entry

#ifdef PHP_WIN32
#define PHP_PHPTest_API __declspec(dllexport)
#elif defined(__GNUC__) && __GNUC__ >= 4
#define PHP_PHPTest_API __attribute__((visibility("default")))
#else
#define PHP_PHPTest_API
#endif

#ifdef ZTS
#include "TSRM.h"
#endif

PHP_MINIT_FUNCTION(PHPTest);
PHP_MSHUTDOWN_FUNCTION(PHPTest);
PHP_RINIT_FUNCTION(PHPTest);
PHP_RSHUTDOWN_FUNCTION(PHPTest);
PHP_MINFO_FUNCTION(PHPTest);

// PHP_FUNCTION 用於定義要導出給php調用的函數名稱，這裏我們定義了3個函數： init_module, test_modul
e, close_module
// PHP_FUNCTION 只用來聲明函數的名稱，置於函數的參數將在cpp中定義

PHP_FUNCTION(init_module);
PHP_FUNCTION(test_module);
PHP_FUNCTION(close_module);

/*
    Declare any global variables you may need between the BEGIN
and END macros here:
ZEND_BEGIN_MODULE_GLOBALS(CSVirusAnalyse)
long  global_value;
char *global_string;
ZEND_END_MODULE_GLOBALS(CSVirusAnalyse)
*/

/* In every utility function you add that needs to use variables
in php_CSVirusAnalyse_globals, call TSRMLS_FETCH(); after declaring other
variables used by that function, or better yet, pass in TSRMLS_CC
after the last function argument and declare your utility function
with TSRMLS_DC after the last declared argument.  Always refer to
the globals in your function as CSGAVIRUSANALYSIS_G(variable).  You are
encouraged to rename these macros something shorter, see
examples in any other php module directory.
*/

#ifdef ZTS
#define PHPTTEST_G(v) TSRMG(PHPTest_globals_id, zend_PHPTTest_globals *, v)
#else
#define PHPTTEST_G(v) (PHPTTest_globals.v)
#endif

```

```
#endif/* PHP_TEST_MAIN_H*/
```

ADVERTISEMENT



編譯Mian.cpp文件:

```
// 聲明以下的宏定義解決在編譯過程中會發生：error C2466：不能分配常量大小為0 的數組的錯誤。
#ifdef PHP_WIN32
#define _STATIC_ASSERT(expr) typedef char __static_assert_t[ (expr)?(expr):1 ]
#else
#define _STATIC_ASSERT(expr) typedef char __static_assert_t[ (expr) ]
#endif

// #include "XXXXX.h" 在以下包含頭文件的前面包含要用到的c++ 的stl的頭文件，或者你自己寫的C++的頭文件。
#include <string>
using namespace std;

extern "C"{
#include "zend_config.w32.h"
#include "php.h"
#include "ext/standard/info.h"
#include "Main.h"
}

// 聲明了擴展庫的導出函數列表
zend_function_entry PHPTest_functions[] = {
    PHP_FE(init_module, NULL)
    PHP_FE(test_module, NULL)
    PHP_FE(close_module, NULL)
    PHP_FE_END
};

zend_module_entry PHPTest_module_entry = {
#if ZEND_MODULE_API_NO >= 20010901
    STANDARD_MODULE_HEADER,
#endif
    "PHPTest",
    PHPTest_functions,
    PHP_MINIT(PHPTest),
    PHP_MSHUTDOWN(PHPTest),
    PHP_RINIT(PHPTest), /* Replace with NULL if there's nothing to do at request start */
    PHP_RSHUTDOWN(PHPTest), /* Replace with NULL if there's nothing to do at request end */
    PHP_MINFO(PHPTest),
#if ZEND_MODULE_API_NO >= 20010901
    "0.1", /* Replace with version number for your extension */
#endif
    STANDARD_MODULE_PROPERTIES
};

ZEND_GET_MODULE(PHPTest);

PHP_MINIT_FUNCTION(PHPTest)
{
    /* If you have INI entries, uncomment these lines
    REGISTER_INI_ENTRIES();
    */
    return SUCCESS;
}

PHP_MSHUTDOWN_FUNCTION(PHPTest)
```

```
{
    /* uncomment this line if you have INI entries
    UNREGISTER_INI_ENTRIES();
    */
    return SUCCESS;
}

PHP_RINIT_FUNCTION(PHPTest)
{
    return SUCCESS;
}

PHP_RSHUTDOWN_FUNCTION(PHPTest)
{
    return SUCCESS;
}

PHP_MINFO_FUNCTION(PHPTest)
{
    php_info_print_table_start();
    php_info_print_table_header(2, "PHPTest support", "enabled");
    php_info_print_table_end();

    /* Remove comments if you have entries in php.ini
    DISPLAY_INI_ENTRIES();
    */
}

// 以下是php導出函數的實現，比如string init_module(string content)
PHP_FUNCTION(init_module)
{
    char *content = NULL;    //
    int argc = ZEND_NUM_ARGS();
    int content_len;
    // 這句話便是導出傳入參數
    if (zend_parse_parameters(argc TSRMLS_CC, "s", &content, &content_len) == FAILURE)
        return;
    if(content)
    {
        // 這裏只是為了測試，直接把傳入值返回去。
        string strRet = content;
        // 返回值
        RETURN_STRING((char*)strRet.c_str(), 1);
    }
    else
    {
        php_error(E_WARNING, "init_module: content is NULL");
    }
}

// 以下是int test_module(string content)函數的實現
PHP_FUNCTION(test_module)
{
    char *content = NULL;
    int argc = ZEND_NUM_ARGS();
    int content_len;

    if (zend_parse_parameters(argc TSRMLS_CC, "s", &content, &content_len) == FAILURE)
```

```

        return;
    if(content)
    {
        int nRet = content_len;
        RETURN_LONG(nRet);
    }
    else
    {
        php_error(E_WARNING, "test_module: &content is NULL");
    }
}

// 以下是 void close_module()函數的實現
PHP_FUNCTION(close_module)
{
    if (zend_parse_parameters_none() == FAILURE) {
        return;
    }
    php_printf("close_module successfully\n");
}

```

ADVERTISEMENT



ok, 編寫完以上的文件後, 編譯一下, 將生成的dll文件, 拷貝到正常工作的php 的ext文件夾下, 並在php.ini上配置, 在extension=php\_zip.dll後面添加extension=PHPTest.dll。然後重啟Apache。

編寫php測試 代碼

```

<?php
echo init_module('test init');
echo'<br>';
//輸出: test init
echo test_module('test_module');
echo'<br>';
close_module();
?>

```

出現問題



由於生成的PHPTest.dll 與PHP安裝環境不一致導致，解決方法（非常重要）

為了解決這個問題走了很多彎路，開始以為是PHP源碼版本的問題，下載了很多個版本都沒成功，浪費了很多時間

解決很簡單：在php\_src\main\config.w32.h文件中增加 **#define PHP\_COMPILER\_ID "VC9"**用VC9編譯

運行結果：

```
test init
11
close_module successfully
```

## 五、註意

1、註意你的頭文件的包含的順序。

將你的頭文件以及Windows和C++的頭文件包含在php頭文件的前面

```
#include "xxx.h" // 你的頭文件
extern "C"{
#include "zend_config.w32.h"
#include "php.h"
#include "ext/standard/info.h"
#include "Main.h"
}
```

2.可能遇到error C2466: 不能分配常量大小為0 的數組

解決方法：

```
在vc的 c:\program files\Microsoft visual studio 8\vc\include\malloc.h 文件中找到:
#define _STATIC_ASSERT(expr) typedef char __static_assert_t[ (expr) ]
將這一行改為:
#ifdef PHP_WIN32
#define _STATIC_ASSERT(expr) typedef char __static_assert_t[ (expr)?(expr):1 ]
#else
#define _STATIC_ASSERT(expr) typedef char __static_assert_t[ (expr) ]
#endif
```

或者直接在你的cpp文件中定義也可以。

2. 如果遇到2019連接錯誤，那麼通常是沒有刪除預處理定義中的宏LIBZEND\_EXPORTS，

## 六、遇到問題



提示找不到php5ts.dll，是因為在這個項目中，在編輯PHPTest.dll的main.cpp文件中調用了之前自己寫的動態庫，如何方法之前的identify.dll庫也成了一個問題，首先在PHPTest工程中添加動態鏈接庫identify.lib，添加方法同php5ts.lib

然後把identify.dll拷貝到aphach安裝目錄下bin文件夾內C:\Apache24\bin 即可完成動態庫的調用。

**Tags:** Windows windows PHP源碼 處理器 二進制

文章來源：



#### 相關文章

- windows下php5擴展開發環境搭建 (/articles/1478584208.html) 2016-11-08
- PHP擴展開發—windows開發環境搭建 (/articles/1478552127.html) 2016-11-08
- 基於PHP7的PHP擴展開發之五（定義常量） (/articles/1490741828.html) 2017-03-29
- 基於PHP7的PHP擴展開發之二（函數的傳參與返回值） (/articles/1490324301.html) 2017-03-24
- Php擴展開發技能圖 (/articles/1476309005.html) 2016-10-13
- 編寫php擴展(二)構建擴展中的函數的說明之PHP擴展開發及內核應用之閱讀筆記（一） (/articles/1476063339.html) 2016-10-10
- PHP擴展開發—windows (/articles/1478532606.html) 2016-11-07
- PHP擴展開發——環境部署 (/articles/1476402618.html) 2016-10-14
- Windows下PHP擴展資源列表及下載地址(官方) (/articles/1498743482.html) 2017-06-29
- 基於PHP7的PHP擴展開發之五（數組的處理） (/articles/1490324411.html) 2017-03-24
- 基於PHP7的PHP擴展開發之四（字符串的處理） (/articles/1490324310.html) 2017-03-24
- 基於PHP7的PHP擴展開發之三（判斷傳入參數的類型） (/articles/1490324304.html) 2017-03-24
- 基於PHP7的PHP擴展開發之一（hello word） (/articles/1490230804.html) 2017-03-23
- PHP擴展開發—LINUX環境 (/articles/1478532603.html) 2016-11-07