

[WGI demo]

The CFS Design of WGI demo Project

[V0.6]

作者：赵平智

2011.08.06

★ 版权所有，保留所有权利 ★

文档修订记录

版本号	变更内容	日期	变更人
V0.2	创建。	20110706	赵平智
V0.6	修改补充。	20110806	赵平智

版权所有，保留所有权利！

目录

1	前言	3
2	函数	3
2.1	LOADPICTUREFILE (加载图片文件)	3
2.2	ISINTER2RECTS (判断两个在同一平面的矩形是否相交)	5
2.3	REFRESH_DCLL_PFIRSTIIRSBDT (刷新 pFIRSTIIRSBDT 双向循环链表)	6
2.4	CALCULATEIRROLL (做好内部矩形滚动前的一些数值计算)	6
2.5	MAKECURRENTBITMAP (生成当前位图)	7
2.6	PAINT (显示当前位图)	8
2.7	REPAINTSUPERIOR (重新绘制上级对象)	9
2.8	ACTIVATE (激活对象)	10
2.9	DEACTIVATE (使对象退出激活状态)	12
3	XXX	14

1 前言

WGI 是用 OIOIC 技术实现的 GUI (图形用户界面) 库, 也是 OIOIC 技术的代表作。
OIOIC 是一种专为 C 语言设计的全新的面向对象编程机制。

WGI 下载地址: <http://code.google.com/p/oic-wgi/downloads/list>

OIOIC 下载地址: <http://code.google.com/p/oioic/downloads/list>

2 函数

2.1 LoadPictureFile (加载图片文件)

* 需求编号:	
* 需求功能:	加载图片文件。
* 函 数:	HBITMAP LoadPictureFile(const TCHAR* adrame)
* 描 述:	加载图片文件, SE 函数。
* 参 数:	adrame- [IN] address and name, 图片地址和名称。
* 返 回:	非 NULL - 加载成功; NULL - 加载失败。
* 定 义 者:	赵平智
* 设 计 者:	赵平智
* 实 现 者:	赵平智
* 使 用 者:	
* 版 本:	V1.0.8
* 更新日期:	20110713
* 备 注:	需要 #include <olectl.h>
* 修改记录:	
* 测试要点:	
设 计	
<pre>HBITMAP LoadPictureFile(const TCHAR* name) { HANDLE hFile; HBITMAP hBmp; DWORD dwSize; DWORD dwRead; HGLOBAL hMemJpeg; LPSTREAM lpStream; OLE_HANDLE hJpegBmp; HRESULT hr; LPPICTURE lpPicture = NULL; void *pMemJpeg; /* Open the file and get the size. */ if((hFile = CreateFile(name, GENERIC_READ, FILE_SHARE_READ,</pre>	

```
NULL, OPEN_EXISTING, 0, NULL)) == INVALID_HANDLE_VALUE)
    return NULL;
if((dwSize = GetFileSize(hFile, NULL)) == 0xFFFFFFFF)
{
    CloseHandle(hFile);
    return NULL;
}

/* Allocate space for file, read it in, and then close the file again. */
if((hMemJpeg = GlobalAlloc(GMEM_MOVEABLE, dwSize)) == NULL)
{
    CloseHandle(hFile);
    return NULL;
}
if((pMemJpeg = GlobalLock(hMemJpeg)) == NULL)
{
    CloseHandle(hFile);
    GlobalFree(hMemJpeg);
    return NULL;
}
if(!ReadFile(hFile, pMemJpeg, dwSize, &dwRead, NULL))
{
    CloseHandle(hFile);
    GlobalFree(hMemJpeg);
    return NULL;
}
CloseHandle(hFile);
GlobalUnlock(hMemJpeg);

/* Create the stream and load the picture. */
if((hr = CreateStreamOnHGlobal(hMemJpeg, TRUE, &lpStream)) != S_OK)
{
    GlobalFree(hMemJpeg);
    return NULL;
}
if(OLELoadPicture(lpStream, dwSize, FALSE, &IID_IPicture, &lpPicture) != S_OK)
{
    GlobalFree(hMemJpeg);
    lpStream->lpVtbl->Release(lpStream);
    return NULL;
}

/* Get the handle to the image, and then copy it. */
if((lpPicture->lpVtbl->get_Handle(lpPicture, &hJpegBmp)) != S_OK)
{
```

```
GlobalFree(hMemJpeg);
lpStream->lpVtbl->Release(lpStream);
lpPicture->lpVtbl->Release(lpPicture);
return NULL;
}
if((hBmp = CopyImage((HANDLE *) hJpegBmp, IMAGE_BITMAP, 0, 0, LR_COPYRETURNORG))
== NULL)
{
GlobalFree(hMemJpeg);
lpStream->lpVtbl->Release(lpStream);
lpPicture->lpVtbl->Release(lpPicture);
return NULL;
}

/* Free the original image and memory. */
GlobalFree(hMemJpeg);
lpStream->lpVtbl->Release(lpStream);
lpPicture->lpVtbl->Release(lpPicture);

return hBmp;
} /* End of LoadPictureFile(). */
```

2.2 IsInter2Rects（判断两个在同一平面的矩形是否相交）

* 需求编号:	
* 需求功能:	判断两个在同一平面的矩形是否相交。
* 函 数:	BOOL IsInter2Rects(const RECT* pSrcRect1, const RECT* pSrcRect2, RECT* pDstRect)
* 描 述:	判断两个平面矩形是否相交（包括内含），SE 函数。
* 参 数:	pSrcRect1 - [IN] 第一个平面多边形。 pSrcRect2 - [IN] 第二个平面多边形。 pDstRect - [OUT] 这个参数可以为 NULL，如果不为 NULL, 输出两个矩形相交的部分（也是矩形），如果两个矩形不相交, 就把这个结构体置空。
* 返 回:	TRUE - 两平面矩形相交，FALSE - 两平面矩形不相交。
* 定 义 者:	赵平智
* 设 计 者:	赵平智
* 实 现 者:	赵平智
* 使 用 者:	
* 版 本:	V1.0.8
* 更新日期:	20110713
* 备 注:	
* 修改记录:	
* 测试要点:	

设 计

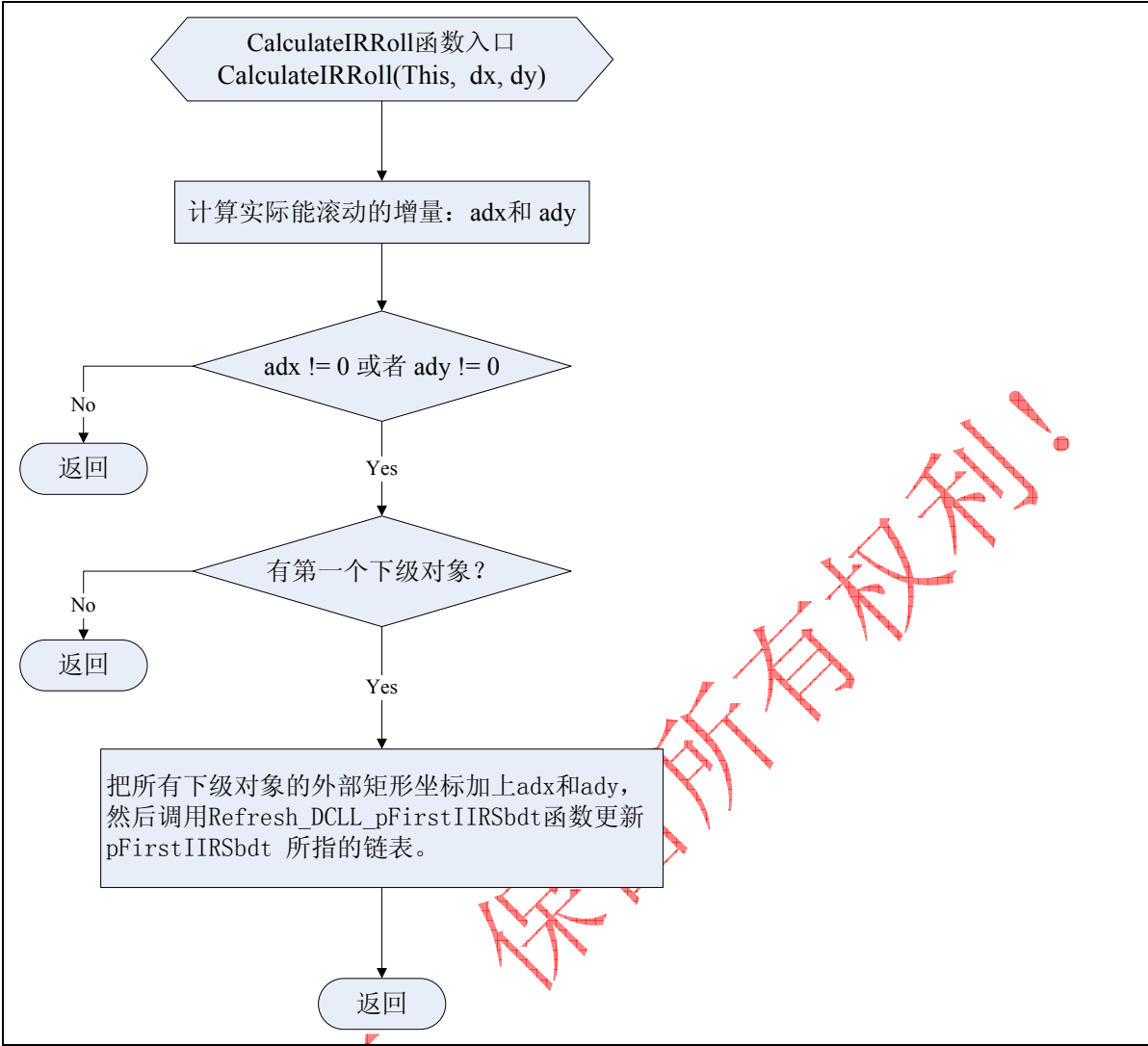
使用 API 函数 IntersectRect。

2.3 Refresh_DCLL_pFirstIIRSbdt (刷新 pFirstIIRSbdt 双向循环链表)

* 需求编号:	
* 需求功能:	刷新 pFirstIIRSbdt 所指的双向循环链表。
* 函 数:	void Refresh_DCLL_pFirstIIRSbdt(OBJECT* This)
* 描 述:	刷新 pFirstIIRSbdt 所指的双向循环链表。VObject 节点函数。
* 参 数:	
* 返 回:	
* 定 义 者:	赵平智
* 设 计 者:	赵平智
* 实 现 者:	赵平智
* 使 用 者:	
* 版 本:	V1.0.8
* 更新日期:	20110713
* 备 注:	DCLL - Doubly Circularly Linked List, 双向循环链表。
* 修改记录:	
* 测试要点:	
设 计	
简单, 略。	

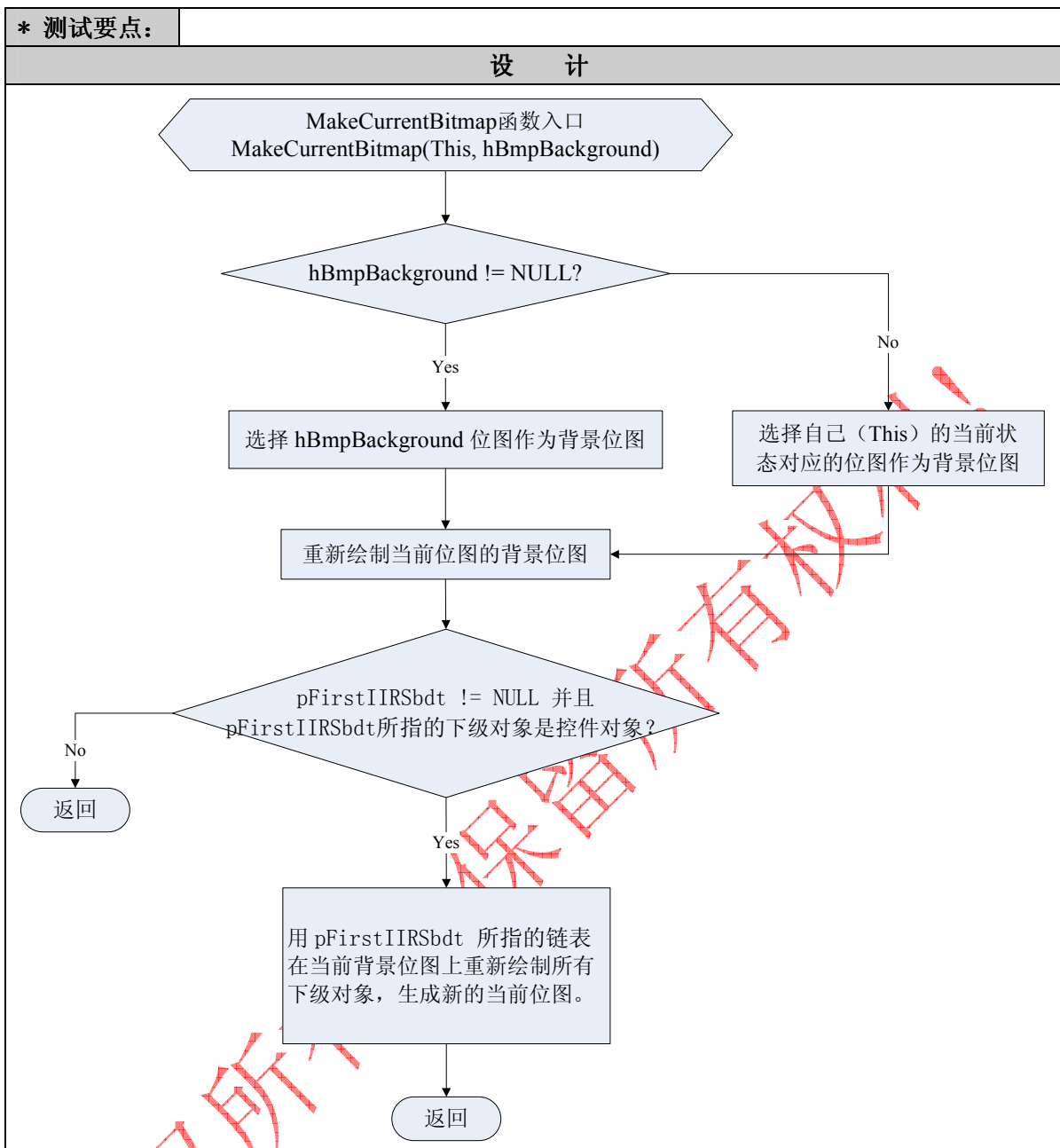
2.4 CalculateIRRoll (做好内部矩形滚动前的一些数值计算)

* 需求编号:	
* 需求功能:	做好内部矩形滚动前的一些数值计算。
* 函 数:	void CalculateIRRoll(OBJECT* This, long dx, long dy)
* 描 述:	做好内部矩形滚动前的一些数值计算。VObject 节点函数。
* 参 数:	dx - [IN] 沿 X 坐标轴的滚动增量, 可以为负值; dy - [IN] 沿 Y 坐标轴的滚动增量, 可以为负值。
* 返 回:	
* 定 义 者:	赵平智
* 设 计 者:	赵平智
* 实 现 者:	赵平智
* 使 用 者:	
* 版 本:	V1.0.8
* 更新日期:	20110713
* 备 注:	IR - Inner Rectangle。
* 修改记录:	
* 测试要点:	
设 计	



2.5 MakeCurrentBitmap（生成当前位图）

* 需求编号:	
* 需求功能:	生成当前位图 EM_VOBJECT. hBmpCurrent。
* 函 数:	static void MakeCurrentBitmap(OBJECT* This, const HBITMAP hBmpBackground)
* 描 述:	生成当前位图 EM_VOBJECT. hBmpCurrent。VObject 节点函数。
* 参 数:	hBmpBackground - [IN] 背景位图，背景位图即没有绘制下级对象的干净位图。
* 返 回:	无
* 定 义 者:	赵平智
* 设 计 者:	赵平智
* 实 现 者:	赵平智
* 使 用 者:	
* 版 本:	V1.0.8
* 更新日期:	20110713
* 备 注:	
* 修改记录:	

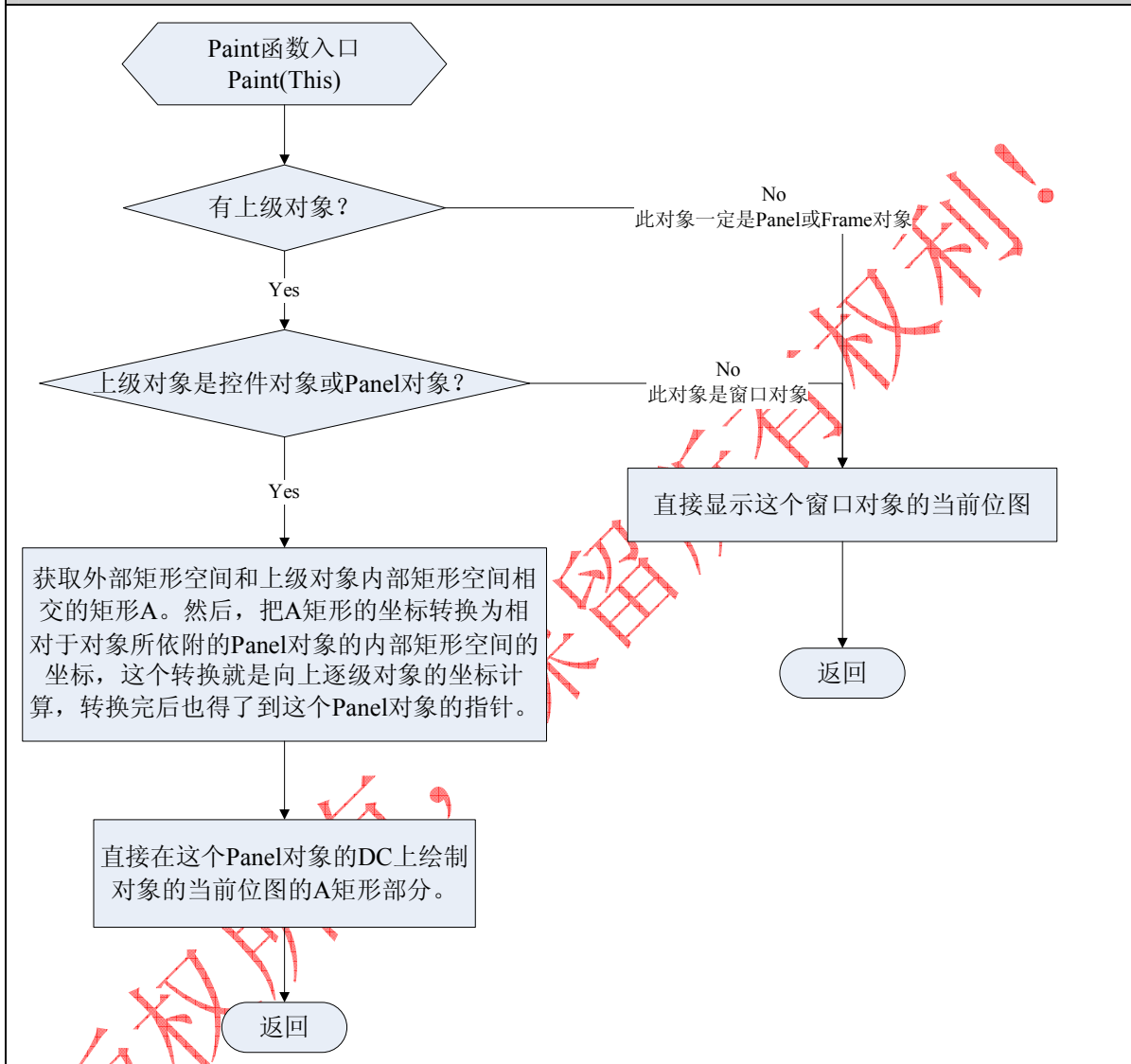


2.6 Paint (显示当前位图)

* 需求编号:	
* 需求功能:	显示当前位图 EM_VOBJECT.hBmpCurrent。
* 函 数:	static void Paint(OBJECT* This)
* 描 述:	显示当前位图 EM_VOBJECT.hBmpCurrent。VObject 节点函数。
* 参 数:	
* 返 回:	无
* 定 义 者:	赵平智
* 设 计 者:	赵平智
* 实 现 者:	赵平智
* 使 用 者:	

* 版本:	V1.0.8
* 更新日期:	20110713
* 备注:	
* 修改记录:	
* 测试要点:	

设 计

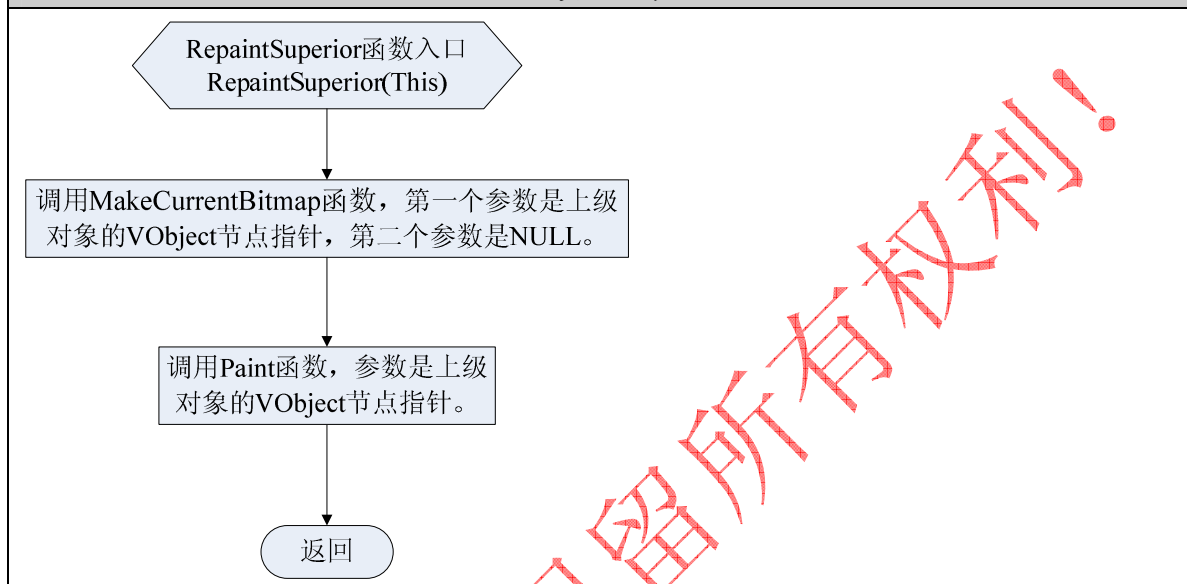


2.7 RepaintSuperior (重新绘制上级对象)

* 需求编号:	
* 需求功能:	重新绘制上级对象。
* 函 数:	static void RepaintSuperior(OBJECT* This)
* 描 述:	重新绘制上级对象。VObject 节点函数。
* 参 数:	
* 返 回:	无
* 定 义 者:	赵平智
* 设 计 者:	赵平智

* 实 现 者:	赵平智
* 使 用 者:	
* 版 本:	V1.0.8
* 更新日期:	20110713
* 备 注:	
* 修改记录:	
* 测试要点:	

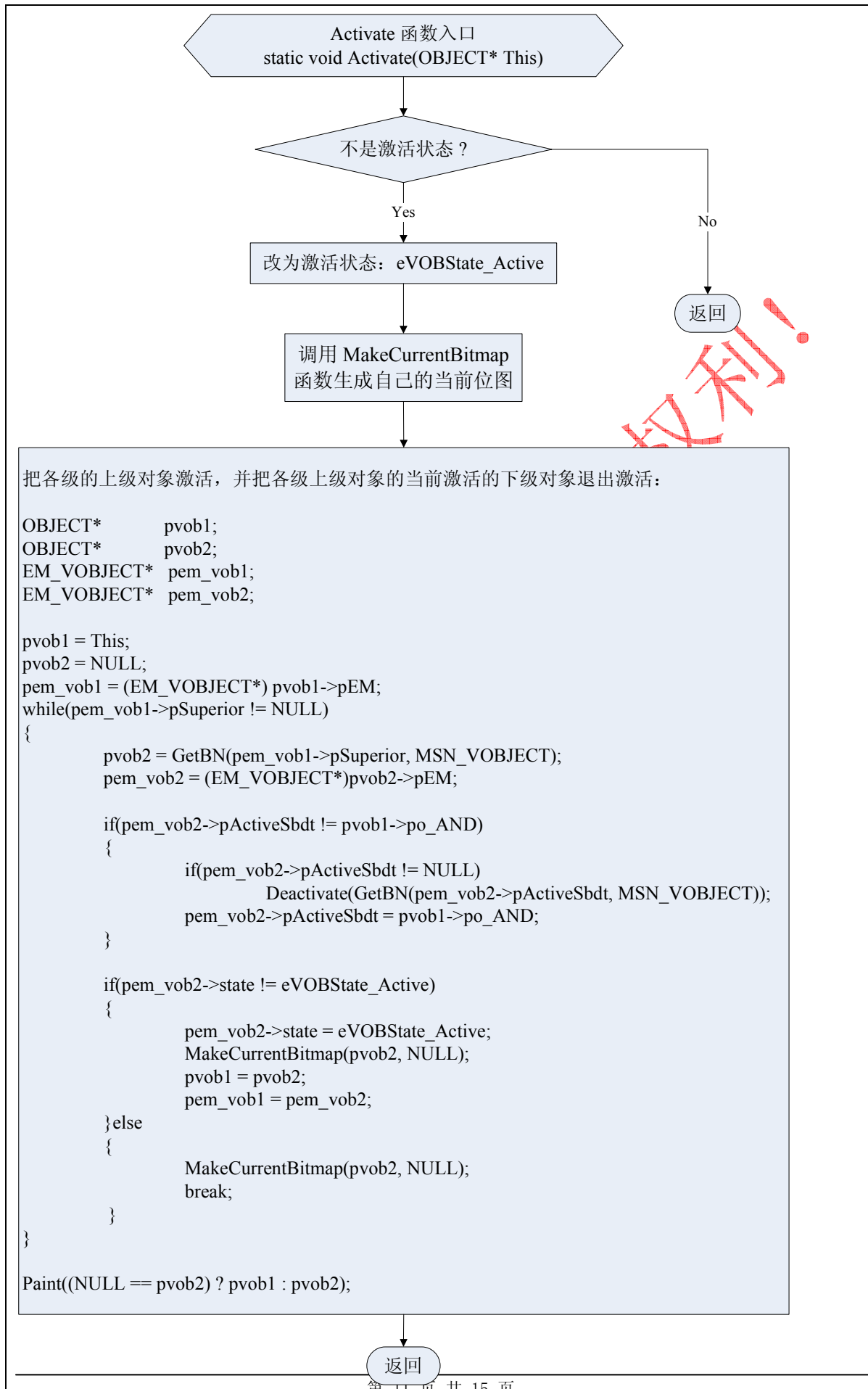
设 计



2.8 Activate（激活对象）

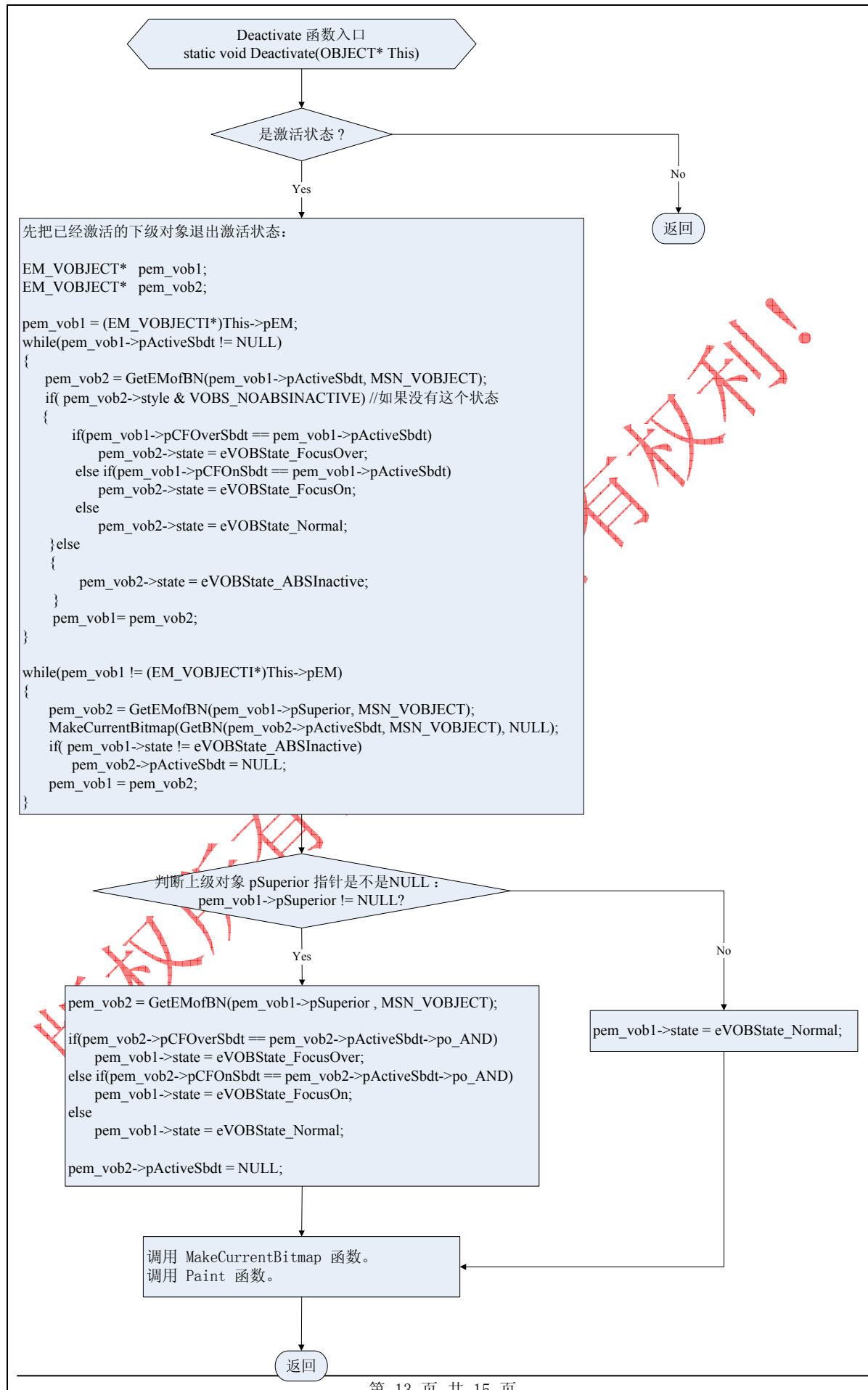
* 需求编号:	
* 需求功能:	激活对象
* 函 数:	static void Activate(OBJECT* This)
* 描 述:	激活对象，MSG_WGI_ACTIVATE 行为调用。VObject 节点函数。
* 参 数:	
* 返 回:	无
* 定 义 者:	赵平智
* 设 计 者:	赵平智
* 实 现 者:	赵平智
* 使 用 者:	
* 版 本:	V1.0.8
* 更新日期:	20110713
* 备 注:	
* 修改记录:	
* 测试要点:	

设 计



2.9 Deactivate (使对象退出激活状态)

* 需求编号:	
* 需求功能:	使对象退出激活状态
* 函 数:	static void Deactivate(OBJECT* This)
* 描 述:	使对象退出激活状态, MSG_WGI_DEACTIVATE 行为调用。VObject 节点函数。
* 参 数:	
* 返 回:	无
* 定 义 者:	赵平智
* 设 计 者:	赵平智
* 实 现 者:	赵平智
* 使 用 者:	
* 版 本:	V1.0.8
* 更新日期:	20110713
* 备 注:	
* 修改记录:	
* 测试要点:	
设 计	



3 xxx

版权所有，保留所有权利！