

'5
 Direct3D Žf f fffff
 μŠ §
 f ffffŠ §
 IDirect3DRM" —fff ffff §
 IDirect3DRMArray §
 IDirect3DRMDeviceArray §
 IDirect3DRMFaceArray §
 IDirect3DRMFrameArray §
 IDirect3DRMLightArray §

IDirect3DRMPickedArray §

IDirect3DRMViewportArray.....	§
IDirect3DRMVisualArray.....	§
IDirect3DRM.....	§
IDirect3DRMAnimation.....	§
IDirect3DRMAnimationSet.....	§
IDirect3DRMDevice.....	§
IDirect3DRMFace.....	§
IDirect3DRMFrame.....	§
IDirect3DRMLight.....	§
IDirect3DRMMaterial.....	§
IDirect3DRMMesh.....	§
IDirect3DRMMeshBuilder.....	§
IDirect3DRMObject.....	§
IDirect3DRMShadow.....	§
IDirect3DRMTexture.....	§
IDirect3DRMUserVisual.....	§
IDirect3DRMViewport.....	§
IDirect3DRMWinDevice.....	§
IDirect3DRMWrap.....	§
“.....	§
—<E.....	§
„‘E.....	§
•’.....	§

Š

xe "Direct3DRMCreate"§

Direct3DRMCreate

HRESULT Direct3DRMCreate(LPDIRECT3DRM FAR * lplpD3DRM);

Direct3DRMffffff,ffffff, ,,

•.....□CE,,,DP_OK□Ž”,,,ff ,•, •’,^—,,,,, Direct3D•Žf f,•’,Ž ,,,,

lplpD3DRM

CE, ,, CE,, —CE,Direct3DRMffff,“,ffff,ffff

Ž Direct3DRMObject

xe "D3DRMColorGetAlpha"§

D3DRMColorGetAlpha

D3DVALUE D3DRMColorGetAlpha(D3DCOLOR d3drmc);

,ffff—‘,Ž“,,

•..... CE,,,ffff’,•, Ž”,,,0,•,□

d3drmc

ffff—‘,Ž“,,□□

Ž□□D3DRMColorGetBlue, D3DRMColorGetGreen, D3DRMColorGetRed

xe "D3DRMColorGetBlue"§

D3DRMColorGetBlue

D3DVALUE D3DRMColorGetBlue(D3DCOLOR d3drmc);

□, —‘,Ž“,,

•..... CE,,, ’,•, Ž”,,,0,•,

d3drmc

—‘,Ž“,,

Ž D3DRMColorGetAlpha, D3DRMColorGetGreen,

D3DRMColorGetRed

xe "D3DRMColorGetGreen"§

D3DRMColorGetGreen

D3DVALUE D3DRMColorGetGreen(D3DCOLOR d3drmc);

,——‘,Ž“,,

•..... □CE,,,—□’,•, Ž”,,,0,•,

d3drmc

```

——‘,Ž“,,
Ž D3DRMColorGetAlpha, D3DRMColorGetBlue, D3DRMColorGetRed

xe "D3DRMColorGetRed"
D3DRMColorGetRed
D3DVALUE D3DRMColorGetRed(D3DCOLOR d3drmc);

, —‘,Ž“,,
•..... Ą,, ’, Ž”,,,0,•,

d3drmc
——‘,Ž“,,
Ž D3DRMColorGetAlpha, D3DRMColorGetBlue, D3DRMColorGetGreen

xe "D3DRMCreateColorRGB"§
D3DRMCreateColorRGB
D3DCOLOR D3DRMCreateColorRGB(D3DVALUE red, D3DVALUE green,
D3DVALUE blue);

—,,,□□—,,, —‘,, RGB , , ,
•..... Ą,,, ,,RGB’,•, Ž”,,,0,•,

red, green, blue
RGB , —‘
Ž D3DRMCreateColorRGBA

xe "D3DRMCreateColorRGBA"§
D3DRMCreateColorRGBA
D3DCOLOR D3DRMCreateColorRGBA(D3DVALUE red, D3DVALUE green,
D3DVALUE blue, D3DVALUE alpha);

—,,,□□—□□□□,,,fff—‘,, RGBA , , ,
•..... Ą,,, ,,RGBA’,•, Ž”,,,0,•,

red, green, blue, alpha
RGBA , —‘
Ž□□D3DRMCreateColorRGB

xe "D3DRMFREEFUNCTION"§
D3DRMFREEFUNCTION
typedef VOID (*D3DRMFREEFUNCTION)(LPVOID lpArg);
typedef D3DRMFREEFUNCTION *LPD3DRMFREEFUNCTION;

```

fff,%o*,,,Š,Ž,ffff fff',<,,
•.....*

lpArg

ffff fff',f f,ffff

• Cffffff f ff,,*, ffff fff, “Ž,fff%o*Š ,',,,,,,□

xe "D3DRMMALLOCFUNCTION"§

D3DRMMALLOCFUNCTION

typedef LPVOID (*D3DRMMALLOCFUNCTION)(DWORD dwSize);

typedef D3DRMMALLOCFUNCTION *LPD3DRMMALLOCFUNCTION;

fff,Š,“,□,Š,Ž,ffff fff',<,,□

•..... E,, Š,“,,,,fff,ffff,*, Ž”,,,0,*,

dwSize

Š,“,,,,fff,‘,,,fff’^,Ž’,,

• Cffffff f ff,,*□□□ffff fff,□“Ž,fffŠ,“Š□,’,,,,,,

xe "D3DRMMatrixFromQuaternion"§

D3DRMMatrixFromQuaternion

void D3DRMMatrixFromQuaternion (D3DRMMATRIX4D mat,

LPD3DRMQUATERNION lpquat);

%“,Ž, —,EŽ,,

•.....*

mat

Š , E,*, EŽ,, —,“,ffff D3DRMMATRIX4DE,”—,,

lpquat

D3DRMQUATERNION ‘“,fff .

xe "D3DRMQuaternionFromRotation"§

D3DRMQuaternionFromRotation

LPD3DRMQUATERNION

D3DRMQuaternionFromRotation(LP3DRMQUATERNION lpquat,

LPD3DVECTOR lpv, D3DVALUE theta);

Ž’,,,Ž,’ , Ž’,,,Š“ ffff”^□,%“, ,

•..... E,, ‘1fff f,,“,,,4Ž —,fff,*, Ž”,,,0,*,

lpquat

LPD3DVECTOR lpaxis, D3DVALUE theta);

Ž',,,,Ž,'□,ffff,%"',,,,□

- □E,,,□'1fff□f,,,□'ffff,•□Ž",,,0,•□

lpr

□—E%o,Ž,Ž,D3DVECTOR □“□

lpv

の D3DVECTOR

lpaxis

の D3DVECTOR

theta

の%o“ Š“

xe "D3DRMVectorScale"§

D3DRMVectorScale

LPD3DVECTOR D3DRMVectorScale(LP3DVECTOR lpd, LPD3DVECTOR

lps,

D3DVALUE factor);

の

- □E, 1

0

lpd

□□□□□ Ž,or

□ŽE□□,,'1,□ff□fff,□,,,□,,'2,,,2",,,,□

xe "D3DRMVectorSubtract"§

D3DRMVectorSubtract

LPD3DVECTOR D3DRMVectorSubtract(LP3DVECTOR lpd, LPD3DVECTOR lps1,

LPD3DVECTOR lps2);

,,,,ffffŠ,Ž,□,□

- 1

0

lpd

D3DVECTOR “

lps1

lps2 ,^,,D3DVECTOR “

lps2

```

lps1 , ^, D3DVECTOR  “
f  ffffŠ

```

```

xe "D3DRMDEVICEPALETTECALLBACK"§
D3DRMDEVICEPALETTECALLBACK
void (*D3DRMDEVICEPALETTECALLBACK)
(LPDIRECT3DRMDEVICE lpDirect3DRMDev, LPVOID lpArg, DWORD
dwIndex,

```

LONG red, Ž‘, ffff □ fff, ‘<, , , □

- ‘, , , , □

```

lpDirect3DRMDev
, , ffff, IDirect3DRMDevice fff □ ffff □
lpArg
f □ ffff Š □ , “, , , ffff □ fff ‘ < , f □ f □
dwIndex
- < , , , ffff □ ffff , ffff fff □
red, green, blue
Ž ‘ , , , ffff □ ffff fff , □ □ - □ □ - ‘ □
ffff , f □ ffff Š □ , Œ , □ , □ □ , Œ ‘ , , , , □    ①

```

```

xe "D3DRMFRAMEMOVECALLBACK"§
D3DRMFRAMEMOVECALLBACK
void (*D3DRMFRAMEMOVECALLBACK)(LPDIRECT3DRMFRAME
lpD3DRMFrame,
LPVOID lpArg, D3DVALUE delta);

```

ff □ ① ff, ffffff, ffffff, “—
, , , , , ff □ f □ f □ f, • □ , , , , , f □ ffff Š
①

-

lpD3DRMFrame

```

□ □ □ □ □ □ □ □ Direc □ ff □ f, ^, • □ , , , □ ^ “ % “ , , , , , — ‘ , , □ Š — ‘ , , , • □ , velocity_of_component ‘
delta, • , , , □ , , , , ^ • , , , - • , “ \ “ , □ “ ^ , ff □ f, “ - , □ ‘ , , , , , □ , , , ffff , □ fff Ž Š, “ -
, , , , □ Ž “ \ “ , □ ff □ f, “ - , “ \ “ , • Š , , □ .
fff □ ffff , □ □ ff □ f, “ -
, , , ff □ f, % \ “ , □ □ , , , , , , □ , , , , □ Š □ Š , , , , □ Ž f ①
(1 f , , , a ffff) ‘ 1 , fff , , , , , , , , , a, — < , , , % “ “ , , , , ffff “ , , , a ‘
delta , “ , , “ • “ , , Ž ff f, “ —    ① a ‘ delta , ff f, “ — , (v
+ (a ‘ delta) ffff, • Š , , , , , v, ‘ , , , Œ , Ž ff f, “ “ “ , Ž , , , , ,

```


ffff, f fff
,, ,Š',fff fff,,ffffff,Œ ,, Ž, ,, ",f ffffŠ ,Œ, ,
Ž□□ID:DeleteUpdateCallback, IDirect3DRMDevice::Update

```
xe "D3DRMUSERVISUALCALLBACK"$  
D3DRMUSERVISUALCALLBACK  
int (*D3DRMUSERVISUALCALLBACK) (LPDIRECT3DRMUSERVISUAL lpD3DRMUV,  
LPVOID lpArg, D3DRMUSERVISUALREASON lpD3DRMUVreason,  
LPDIRECT3DRMDEVICE lpD3DRMDev, LPDIRECT3DRMVIEWPORT lpD3DRMview);
```

f□f∅∅<, ,

-

lpD3DRMUVreasonfff f,D3DRMUSERVISUAL_CANSEE,Ž',, f f f
ffff ffffff,ff f f“,%oŽ ‘, , , TRUE,•, , , , ,FALSE,•,
lpD3DRMUVreason fff f,D3DRMUSERVISUAL_RENDER,
fff'<,' , , , , ,TRUE,•, , , ^', ,

lpD3DRMUV
Direct3DRMUserVisual
lpArg

lpD3DRMUVreason

D3DRMUSERVISUALREASON □□□∅□□□ □□
,□f, ff□f□f“,%oŽ□', , ,□ffff□fff,TRUE,•, □, , □□□ffff□fff, □lpD3DRMview
ffff□f, Ž' , , , ffff, Ž-, , □

D3DRMUSERVISUAL_RENDER

ffff□fff, □f□f□fffff□fffff, ffffff, , , , , , □, □□□ffff□fff, □lpD3DRMDevff
f□f, Ž' , , , ffff, Ž, □

lpD3DRMDev

Direct3DRMUserVisual ff,ffffff, , , , Ž, , , Direct3DRMDeviceffffff

lpD3DRMview

Direct3DRMUserVisual ffffff,%oŽ ‘, , , , Œ', , , , Ž, , , Direct3DRMViewport
ffffff

ffff,f□ffffŠ□,Œ,□,□□,
' ,fff□fff, , , ffffff,Œ□, □Ž, □□□, , , □”, f□ffffŠ□,Œ,□,
IDirect3DRMUserVisual::Init

xe "D3DRMWRAPCALLBACK"\$

D3DRMtor,

int* lpU, int* lpV, LPD3DVECTOR lpD3DRMVA, LPD3DVECTOR lpD3DRMVB,
LPVOID lpArg);

,,f□ffffš□,ff□f,,,,,□
IDirect3DRM"-fff□ffff
"-fff□ffff,-,,□ffff□ffff,□ffff□ffff,"-fff□f%,,š',fff□f`□,□-,"-,,,,,,□^%
,"-fff□ffff,ž-,,,,,,,□
IDirect3DRMArray

IDirect3DRMDeviceArray

IDirect3DRMFaceArray

IDirect3DRMFrameArray

IDirect3DRMLightArray

IDirect3DRMPickedArray

IDirect3DRMViewportArray

IDirect3DRMVisualArray

IDirect3DRMArray

IDirect3DRMArrayfff□ffff,ffffff,ff□

□□ □□□□

fff,'□ž,,□,,fff□ffff,,□,,,"-ffffff,——

,,□,,ffffrrayfff□ffff,GetSizeffff,ff□f,,□

,,,COMfff□ffff,"-

□IDirect3DRMArrayfff□ffff,IUnknownfff□ffff□ffff,□,,□,,fff□ffff,ž,,
ffff,ff□f,,□

AddRef

QueryInterface

Release

xe "IDirect3DRMArray\:\:GetSize"§

IDirect3DRMArray::GetSize

DWORD GetSize();

Direct3DRMArray fffffff,fff, fffffff'^,Ž“,,

- f

IDirect3DRMDeviceArray

ffff□ffffff,ff□f%,,,,□IDirect3DRMDeviceArrayfff□ffff□ffff,ffffff,□Š”,,,,Š—

,□□IDirect3DRMDevice, IDirect3DRMDeviceArray fff□ffff□,ž□,,,□

IDirect3DRMDeviceArray fff□ffff,□ž, ff

GetElement

GetSize

IDirect3DRMDeviceArray ◊ COM

IUnknown

f ffff,Œ ,, IUnknownfff ffff,ž,,,ffff,ff f,,

AddRef

QueryInterface

Release

lpD3DRMDevice

IDirect3DRMDevice *fff ffff,,ffff*

DRMDeviceArray::GetSize

DWORD GetSize();

Direct3DRMDeviceArray *ffffff,š,,-`ž",,□*

- —□,□

IDirect3DRMFaceArray

IDirect3DRMFaceArray *fff□ffff,ffff,——*
,,□,,ffff,□IDirect3DRMFaceArrayfff□ffff□ffff,fffff,,□š”,,,,š—
,□□IDirect3DRMFace,IDirect3DRMFaceArrayfff□ffff □,ž□,,,
IDirect3DRMFaceArrayfff ffff, ž,ffff,ff f,,

GetElement

GetSize

IDirectfff□ffff□ffff,Œ□,,□IUnknownfff□ffff,□ž,,,,ffff,ff□f,,□
AddRef

QueryInterface

Release

,,□IDirect3DRMFaceArrayfff□ffff,□IDirect3DRMObjectfff□ffff,,ž,ffff,Œ
□,,□
AddDestroyCallback

Clone

DeleteDestroyCallback

GetAppData

GetClassName

GetName

SetAppData

SetName

Direct3DRMFaceArrayfffff, IDirect3DRMMeshBuilder::GetFacesfff, Ć, ,
 ,,Ž“,,

xe "IDirect3DRMFaceArray\:\:GetElement"\$

IDirect3DRMFaceArray::GetElement

HRESULT GetElement(DWORD index, LPDIRECT3DRMFACE *
 lpD3DRMFace);

Direct3DRMFaceArray Ž“,,

• □Ć,,D3DRM_OK□Ž”,,,ff□,•,□•’,^—
 ,,,,DRMFace

IDirect3DRMFacefff□ffff,,ffff□

xe "IDirect3DRMFaceArray\:\:GetSize"\$

IDirect3DRMFaceArray::GetSize

DWORD GetSize();

Direct3DRMFaceArrayfffff,š,,,—□,ž“,,,□

• —□,•,□

IDirect3DRMFrameArray

ff f fffffff,ff f%o IDirect3DRMFrameArray 0
,,ffffff, IDirect3DRMFrameArrayfff ffff ffff,ffffff,, Š”,,,Š
—, IDirect3DRMFrame,IDirect3DRMFrameArrayfff ffff ,Ž ,,,,
IDirect3DRMFrameArrayfff fff ff f,,
GetElement

GetSize

IDirect3DRMFrameArray ffff, ,,,COMfff ffff,“—
IUnknownfff ffff ffff,€ ,, IUnknown
,,,ffff,ff f,,
AddRef

QueryInterface

Release



,, IIDirect3DRMFrameArrayfff fddDestroyC
allback

Clone

DeleteDestroyCallback

GetAppData

GetClassName

GetName

SetAppData

SetName

Direct3DRMFrameArrayfffff, IDirect3DRMPickedArray::GetPick,,, IDirect3DRMFrame::GetChildren

xe "IDirect3DRMFrameArray\:\:GetElement"§

IDirect3DRMFrameArray::GetElement

HRESULT GetElement(DWORD index, LPDIRECT3DRMFRAME * lpD3DRMFrame);

Direct3DRMFrameArrayfffff, Ž' ,,, —‘

- CE,,,D3DRM_OK • , •', ^—
- ,,,, Direct3D•Žf f,•' ,Ž ,,,,

index

lpD3DRMFrame

IDirect3DRMFrame fff ffff,,ffff,Ž,Ž,ffff

Direct3DRMFrameArrayfffff,Š,,,,—‘□,Ž",□

- —‘□,•,□

IDirect3DRMLightArray

CEfffff,ff f%,,,, IDirect3DRMLightArrayfff ffff,ffff,—
,, ,ffff, IDirect3DRMLightArrayfff ffff ffff,ffffff,, Š”

IDirect3DRMLight IDirect3DRMLightArrayfff ffff ,Ž ,,,,

IDirect3DRMLightArrayfff ffff, Ž,ffff,ff f,,

GetElement

GetSize

の

*IDirect3DRMLightArray*fff□ffff,□,,,COMfff□ffff,
*IUnknown*fff□ffff□ffff,□E□,,□**IUnknown**fff□ffff,Ž,,,ffff

AddRef

QueryInterface

Release

,,, IDirect3DRMLightArray fff ffff, IDirect3DRMObject fff ffff,, Ž, ffff, Œ, ,

AddDestroyCallme

GetName

SetAppData

SetName

```
Direct3DRMFrameArray fffffff,
IDirect3DRMFrame::GetLights ffff, Œ, , , , , Ž", , ,

xe "IDirect3DRMLightArray\:\:GetElement"$
IDirect3DRMLightArray::GetElement
HRESULT GetElement(DWORD index, LPDIRECT3DRMLIGHT * lpD3DRMLight);

Direct3DRMLightArray fffffff, Ž', , , -\, Ž", ,
• D3DRM_OK
Direct3D の
```

index

SetAppData

SetName

Direct3DRMPickedArray ffffffff, IDirect3DRMViewport::Pick

xe "IDirect3DRMPickedArray\:\:GetPick"§

IDirect3DRMPickedArray::GetPick

HRESULT GetPick(DWORD index, LPDIRECT3DRMVISUAL * lpVisual,
LPDIRECT3DRMFRAMEARRAY * lpFrameArray,
LPD3DRMPICKDESC lpD3DRMPickDesc);

Ž',,,,fff,,,,CE ,,Direct3DRMVisual,Direct3DRMFrame ffffffff,Ž“,,,
• ,,D3DRM_OK Ž”,,,ff ,•, •,^—
,,,, Direct3D•Žf f,•' ,Ž ,,,,

index

Ž•,, fff”—,ffffff

lpVisual

Ž',,,,fff,Š~•,,,,Direct3DRMVisual ① ffff,ffff

lpFrameArray

Ž',,,,fff,Š~•,,,,Direct3DRMFrameArray ffffffff,ffff,ffff

lpD3DRMPickDesc

fff,^~ - Ž“,,ffffff,ff fŽ•Ž,Ž’,,D3DRMPICKDESC “,,ffff

Ž IDirect3DRMViewport::Pick

xe "IDirect3DRMPickedArray\:\:GetSize"§

IDirect3DRMPickedArray::GetSize

DWORD GetSize();

Direct3DRMPickedArray ffffffff,Š,,,,—‘ ,Ž“,,,

•

DeleteDestroyCallback

GetAppData

GetClassName

GetName

SetAppData

SetName

Direct3DRMViewportArray

IDirect3DRM::CreateFrame

xe "IDirect3DRMViewportArray\:\.GetElement"§

IDirect3DRMViewportArray::GetElement

HRESULT GetElement(DWORD index, LPDIRECT3DRMVIEWPORT * lpD3DRMViewpor, □

- □E,,,D3DRM_OK□ž",,,,ff□,•,□•',^-,,,,,,□□Direct3D•žf□f,•'□,ž□,.,.,.□

index

"—\□

lpD3DRMViewpor

IDirect3DRMViewpor fff□ffff,ž,ž,ffff,,ffff□

xe "IDirect3DRMViewporArray\:\:GetSize"\$

IDirect3DRMViewporArray::GetSize

DWORD GetSize();

Direct3DRMViewporArray

-

IDirect3DRMVisualArray

ffff ffffff,ff f%o

,,, IDirect3DRMVisualArrayfff ffff,ffff,—,∅

IDirect3DRMVisualArrayfff ffff ffff,ffffff,,, Š",,,,,Š—,

IDirect3DRMVisual IDirect3DRMVisualArrayfff ffff ,ž ,,,,

GetElement

GetSize

IDirect3DRMVisualArrayfff□ffff,□,,,COMfff□ffff,“—

IUnknownfff□ffff□ffff,☉□,,□IUnknown ∅∅

AddRef

QueryInterface

Release

IDirect3DRMVisualArray IDirect3DRMObject の
 AddDestroyCallback

Clone

DeleteDestroyCallback

e

IDirect3DRMVisualArrayfffff, IDirect3DRMFrame::GetVisualsfffff, E, □, , , , , , Ž", , , , □

```

xe "IDirect3DRMVisualArray\:\:GetElement"$
IDirect3DRMVisualArray::GetElement
HRESULT GetElement(DWORD index, LPDIRECT3DRMVISUAL *
lpD3DRMVisual);
  
```

Direct3DRMVisualArrayfffff

- D3DRM_OK
Direct3D の

index

lpD3DRMVisual

IDirect3DRMVisual fff ffff,Ž,Ž,ffff

```
xe "IDirect3DRMVisualArray\:\:GetSize"§
IDirect3DRMVisualArray::GetSize
DWORD GetSize();
```

Direct3DRMVisualArray ffffff,Š,,,,—‘ ,Ž“ ,,
•

```
IDirect3DRM
IDirect3DRM fff, □Direct3DRMfffff, □□, □ffff□f
  の
ff, □IDirect3DRMfff□ffff,ffff,fffff,, ,□Š”,,,,Š—, □ IDirect3DRM
IDirect3DRM の
```

CreateAnimation

CreateAnimationSet

CreateDevice

CreateDeviceFromClipper

CreateDeviceFromD3D

CreateDeviceFromSurface

GetDevices

EnumerateObjects

CreateFace

CreateFrame

CreateLight

CreateLightRGB

CreateMaterial

CreateMesh

CreateMeshBuilder

の

CreateObject

CreateUserVisual

GetNamedObject

Load

Tick

AddSearchPath

GetSearchPath

SetSearchPath

CreateShadow

Cree

LoadTextureFromResource

SetDefaultTextureColors

SetDefaultTextureShades

fff
CreateViewport

fff
CreateWrap

IDirect3DRMfff, , , , COMfff, “–
IUnknownfff, E, , IUnkown のの

AddRef

QueryInterface

Release

IDirect3DRM COM

Direct3DRMCreate の

```
xe "IDirect3DRM\:\AddSearchPath"§
IDirect3DRM::AddSearchPath
HRESULT AddSearchPath(LPCSTR lpPath);
```

```
CE,fffCEff,CE,ffffff,ff
• CE,,D3DRM_OK Ž",ff ,• ,•,^—
,,,, Direct3D•Žf f,• ,Ž ,,,
```

lpPath

```
CE ,CE ff,%o,,ff,Ž',, NULL, ,•Ž—,,fff
```

```
Windows,,ff,ffff□;□,□,fffff,fff,,,,,,,,□
IDirect3DRM::SetSearchPath
```

```
xe "IDirect3DRM\:\CreateAnimation"§
```

```
IDirect3DRM::CreateAnimation
```

```
HRESULT CreateAnimation(LPDIRECT3DRMANIMATION *
lpD3DRMAnimation);
```

○ Direct3DRMAnimation

• □□□□□ **D3DRM_OK**○Direct3D ○
□□□□□□,,tion **fff**□**ffff**,Ž,Ž,**ffff**,,**ffff**□

```
xe "IDirect3DRM\:\CreateAnimationSet"§
IDirect3DRM::CreateAnimationSet
HRESULT CreateAnimationSet (LPDIRECT3DRMANIMATIONSET *
lpD3DRMAnimationSet);
```

```
<,Direct3DRMAnimationSetffffff,□□,,□
```

• □CE,,D3DRM_OK□Ž",, Direct3D

```
lpD3DRMAnimationSet
```

```
IDirect3DRMAnimationSet
```

```
fff,Ž,Ž,ffff,,ffff
```

```
xe "IDirect3DRM\:\CreateDevice"§
```

```
IDirect3DRM::CreateDevice
```

HRESULT CreateDevice(DWORD dwWidth, DWORD dwHeight,
LPDIRECT3DRMDEVICE* lpD3DRMDevice);

Windows ,,,,,,

xe "IDirect3DRM\:\:CreateDeviceFromClipper"§

IDirect3DRM::CreateDeviceFromClipper

HRESULT CreateDeviceFromClipper(LPDIRECTDRAWCLIPPER

lpDDClipper,

LPGUID lpGUID, int width, int height,

LPDIRECT3DRMDEVICE * lpD3DRMDevice);

Ž' ,,DirectDrawClipper

Direct3DRM Windows

- D3DRM_OK
Direct3D の

lpDDClipper

DirectDrawClipper

lpGUID

GUIDのの NULL

width , height

lpD3DRMDevice

IDirect3DRMDevice

の

lpGUID NULL ,

D3DPRIMCAPS

D3DPCMPCAPS_LESSEQUAL

D3DPMISCCAPS_CULLCCW

D3DPRASTERCAPS_FOGVERTEX

D3DPSHADECAPS_ALP

HAFLATSTIPPLED

D3DPTADDRESSCAPS_WRAP

D3DPTBLENDCAPS_COPY | D3DPTBLENDCAPS_MODULATE

D3DPTEXTURECAPS_PERSPECTIVE |
 D3DPTEXTURECAPS_TRANSPARENCY
 D3DPTFILTERCAPS_NEAREST

*f ffff ffff,☉,,,,,, ffff fff ,ffffff ffff,f f,, ,,fffff ',fff,ff
 f,,“,—,Ž,ffff fff, lpGUID,NULL,Ž’,‘,,,ffff,—‘,,,,,*

xe "IDirect3DRM\:\:CreateDeviceFromD3D"§

IDirect3DRM::CreateDeviceFromD3D

HRESULT CreateDeviceFromD3D(LPDIRECT3D lpD3D,

**LPDIRECT3DDEVICE lpD3DDev, LPDIRECT3DRMDEVICE *
 lpD3DRMDevice);**

Ž’,,,Direct3Dffffff,Ž,, Direct3DRM Windowsffff,

- D3DRM_OK
 Direct3D の

lpD3D

Direct3D
lpD3DDev
Direct3D
lpD3DRMDevice

IDirect3DRMDevice の

xe "IDirect3DRM\:\:CrFromSurface(LPGUID lpGUID, LPDIRECTDRAW lpDD,

```
LPDIRECTDRAWSURFACE lpDDSBack,  
LPDIRECT3DRMDEVICE * lpD3DRMDevice);
```

Ž',,, DirectDrawf□fff,,□ffffff,-,,Windowsffff,□□,,□
• □E,,D3DRM_OK□Ž"„ff□,•□'ˆ—,,,,□□Direct3D•Žf□f,•□,Ž□,,,,□

lpGUID
Ž' , , ffff□ffff,ff□ff□ GUID
NULL

lpDD

DirectDraw の DirectDraw

lpDDSBack
fff ffff,•,DirectDrawSurfaceffffff
lpD3DRMDevice
E, ,, E,,,,,IDirect3DRMDevicefff ffff,Ž,Ž,ffff,,ffff

xe "IDirect3DRM\:\:CreateFace"§

IDirect3DRM::CreateFace

```
HRESULT CreateFace(LPDIRECT3DRMFACE * lpD3DRMFace);
```

IDirect3DRMFace

• E,, D3DRM_OK
Direct3D•Žf f,•' ,Ž ,,,

lpD3DRMFace

E, ,, E,,,,,IDirect3DRMFacefff ffff,Ž,Ž,ffff,,ffff

D3DVALUE vGreen, D3DVALUE vBlue, LPDIRECT3DRMLIGHT* lpD3DLight

xe "IDirect3DRM::CreateMesh"§

IDirect3DRM::CreateMesh

HRESULT CreateMesh(LPDIRECT3DRMMESH* lpD3DRMMesh);

-, Ž, ,, □, , ffff□fffff, □□, , □ffff, ff□f, %,,,,, •Ž, ,, □

- □E,,,D3DRM_OK □Ž" , , ff□, •, □•' , ^—,, Direct3D の

lpD3DRMMesh

E, , , , IDirect3DRMMeshfff ffff, Ž, Ž, ffff, ffff

xe "IDirect3DRM::CreateMeshBuilder"§

IDirect3DRM::CreateMeshBuilder

HRESULT CreateMeshBuilder(LPDIRECT3DRMMESHBUILDER*
lpD3DRMMeshBuilder);

□, , ffff□

- E, , , D3DRM_OK
Direct3D•Žf f, •' , Ž , , ,

lpD3DRMMeshBuilder

E, , , E, , , IDirect3DRMMeshBuilderfff ffff, ffff, Ž, Ž, ffff

xe "IDirect3DRM::CreateObject"§

IDirect3DRM::CreateObject

HRESULT CreateObject(REFCLSID rclsid, LPUNKNOWN pUnkOuter,
REFIID riid, LPVOID FAR* ppv);

- D3DRM_OK
Direct3D の

rclsid

の

fff,ff,f,ff,^—
,,,, Direct3D•Žf,f,•,Ž,.,.,

lpImage

ffff,f,f,•,D3DRMIMAGE``
lpD3DRMTexture
•,.,.,•,.,.,., IDirect3DRMTexturefff,fff,ž,ž,fff,.,fff
ffff,fffff,.,., Direct3DRM,fff,ff,.,fff,.,fff,f,š~•,.,fff,—
,.,.,fff,fff,fff,fff,fff,fff,ž,—,.,.,.

xe "IDirect3DRM\:\:CreateTextureFromSurface"š
IDirect3DRM::CreateTextureFromSurface

HRESULT CreateTextureFromSurface(LPDIRECTDRAWSURFACE lpDDS,
*LPDIRECT3DRMTEXTURE * lpD3DRMTexture);*

Ž',., DirectDrawf fff,ffff, .,
• ••,D3DRM_OKŽ",fff,•,•,^—,.,., Direct3D•Žf,f,•, Ž,

lpDDS

DirectDrawSurfacefffff

lpD3DRMTexture

•,.,.,•,.,.,., IDirect3DRMTextureff

xe "IDirect3DRM\:\:CreateUserVisual"š
IDirect3DRM::CreateUserVisual

HRESULT CreateUserVisual(D3DRMUSERVISUALCALLBACK fn,

LPVOID lpArg,

LPDIRECT3DRMUSERVISUAL *

l'fff'<,fff,—,fffff,.,

• ••,D3DRM_OKŽ",fff,•,•,^—,.,., Direct3D•Žf,f,•, Ž,.,.,

fn

fff'fff'<, D3DRMUSERVISUALCALLBACKfffš

lpArg

f'fffš, ",fff'fff'<,ff

lpD3DRMUV

•,.,.,•,.,., IDirect3DRMUserVisual f

xe "IDirect3DRM\:\:CreateViewport"š

IDirect3DRM::CreateViewport

HRESULT CreateViewport(LPDIRECT3DRMDEVICE lpDev,

fff, zŽ
ux, uy, uz
fff, yŽ
ou, ov
ffff, E
su, sv
ffff, ŽE
lpD3DRMWrap

IDirect3DRMWrap

fff, Ž, Ž, ffff, ffff

Ž IDirect3DRMWrap

xe "IDirect3DRM\:\EnumerateObjects"§

IDirect3DRM::EnumerateObjects

HRESULT EnumerateObjects(D3DRMOBJECTCALLBACK *func*, LPVOID
lpArg);

ffff, Direct3DRMfffff, ,, *func*

- E,, D3DRM_OK Ž",,, ff ,•, •', ^—
,,,, Direct3D•Žf f,•' ,Ž ,,,

func

ŠDirect3DRMObjectffff □ □

lpArg

f □ fffffŠ □, " ,,, □ fffff □ fff' <, f □ f □

xe "IDirect3DRM\:\GetDevices"§

IDirect3DRM::GetDevices

HRESULT GetDevices(LPDIRECT3DRMDEVICEARRAY* lpDevArray);

ffff, □□, ,,, , Direct3DRM ffff, •, □

- □E,, D3DRM_OK □Ž", ,

Direct3D

lpDevArray

Direct3DRMffff, "—

, Ž, Ž, ffff Direct3DRMDeviceArrayfffff, ,,, , •, IDirect3DRMDevic
eArrayfff ffff ,Ž ,,,

xe "IDirect3DRM\:\GetNamedObject"§

IDirect3DRM::GetNamedObject

```
HRESULT GetNamedObject(const char * lpName,
LPDIRECT3DRMOBJECT* lplpD3DRMObject);
```

- D3DRM_OK

lpName
 lplpD3DRMObject

Direct3DRMObject の

"IDirect3DRM\:\:GetSearchPath"§

IDirect3DRM::GetSearchPath

```
HRESULT GetSearchPath(DWORD * lpdwSize, LPSTR lpszPath);
```

の

- D3DRM_OK

lpdwSize

NULL

lpszPath

NULL の NULL

lpdwSize

の

IDirect3DRM:T Load(LPVOID lpvObjSource, LPVOID lpvObjID,

```
LPIID * lplpGUIDs, DWORD dwcGUIDs, D3DRMLOADOPTIONS d3drmLOFlags,
D3DRMLOADCALLBACK d3drmLoadProc, LPVOID lpArgLP,
D3DRMLOADTEXTURECALLBACK d3drmLoadTextureProc, LPVOID lpArgLTP,
LPDIRECT3DRMFRAME lpParentFrame);
```

fffff, f, ,

- D3DRM_OK

lpvObjSource

f, , fffff, f, , ffff d3drmLOFlags fff, f, f, fff, , , ffff


```
xe "IDirect3DRM\:\LoadTextureFromResource"§
IDirect3DRM::LoadTextureFromResource
HRESULT LoadTextureFromResource(HRSRC rs,
    LPDIRECT3DRMTEXTURE * lplpD3DRMTexture);
```

Ž',,,ff f,,ffff,f f,,

- Ą,,D3DRM_OK Ž",,,ff ,*, *',^—
,,,, Direct3D•Žf f,•' ,Ž ,,,

rs

ff f,ffff

lplpD3DRMTexture

Direct3DRMTextureffff,,ffff

```
xe "IDirect3DRM\:\SetDefaultTextureColors"§
IDirect3DRM::SetDefaultTextureColors
HRESULT SetDefaultTextureColors(DWORD dwColors);
```

Direct3DRMTexture•',^—
,,,,,□□ Direct3D•Žf □ f,•' □,Ž □ ,,,□

dwColors

□,□□

„ffff,□IDirect3DRM::CreateTextureffff,\,ą,□,,,,,,ffff,□,“%
,,,□,,,□,,,ffff,\,„ą,,,□

```
xe "IDirect3DRM\:\SetDefaultTextureShades"§
IDirect3DRM::SetDefaultTextureShades
HRESULT SetDefaultTextureShades(DWORD dwShades);
```

Direct3DRMTexture Ž—,,ffff,%o%, ',,

- Ą,,D3DRM_OK Ž",,,ff ,*, *',^—
,,,, Direct3D•Žf f,•' ,Ž ,,,

dwShades

%o%,

„ffff, IDirect3DRM::CreateTextureffff,‘,Œ, ,,,,,,,fffff,%o%,”%o
,, ,fffff,‘,’,Œ%o,,

xe "IDirect3DRM:\.SetSearchPath"§
IDirect3DRM::SetSearchPath
HRESULT SetSearchPath(LPCSTR lpPath);

ffffŒ ff, ’,,

pPath

Œ ff,, ,’,,ff,Ž, □ NULL, □,,•Ž—□

fffff,Œ ff,Š•□D3DPATH,’,Ž,□,,□,Š•□,□’,,,,,□□□Œfff,<,,,□ffff, f□ff, ,,,□f
fff,, fffffffffff,’,□Ž,Œfff’,,,,,,fffff,ffff,□
Ž□□IDirect3DRM::GetSearchPath

xe "IDirect3DRM:\.Tick"§
IDirect3DRM::Tick
HRESULT Tick(D3DVALUE d3dvalTick);

Direct3DRM
ffff f ff f, , ,ffff,Œ, ,,,, ,,,%“ff f,^,Œ ,“ ‘ ,,,,,, ”
f f,Œ ,fff,ffffff,, , Š~,f ffffŠ “ ,,,,Œ, ,,, ,ffff, ffff
ff ffff,Š—,,,, Œ,•
• D3DRM_OK f□,•,□•,^ Direct3D の

d3dvalTick

IDirect3DRMFrame::SetRotation IDirect3DRMFrame::SetVelocity
の

のののの

IDirect3DRMAnimation

IDirect3DRMAnimationfff□ffff.ffff,——
,,□^□Œ,□fff

IDirect3DRMAnimation ののIDirect3DRMAnimation
IDirect3DRMAnimationSet

IDirect3DRMAnimation のの

AddPositionKey

AddRotateKey

AddScaleKey

DeleteKey

①

SetFrame

SetTime

GetOptions

SetOptions

IDirect3DRMAnimation COM
IUnknown

IUnknown
①①

AddRef

QueryInterface

Release

IIDirect3DRMAnimation

IIDirect3DRMAnimation
AddDestroyCallback

GetAppData

GetClassName

GetName

SetAppData

IIDirect3DRMObject

Clone

DeleteDestroyCallback

SetName

Direct3DRMAnimation::CreateAnimation
IIDirect3DRMAnimation

IIDirect3DRMAnimation
AddPositionKey

IDirect3DRMAnimation::AddPositionKey

HRESULT AddPositionKey(D3DVALUE rvTime, D3DVALUE rvX,
D3DVALUE rvY, D3DVALUE rvZ);

- D3DRM_OK

Dire

ct3D•Žf f,•' ,Ž ,,,

rvTime

fff fff,;~f , ',,,,ŽŠ 0,Œ
“,,”^,~;ŽŠ,Ž',,,,

,, ① rvTime 99 rvTime 49

rvX, rvY, rvZ

3D

IDirect3DRM

Animation::DeleteKey

xe "IDirect3DRMAnimation\
\:AddRotateKey"§

IDirect3DRMAnimation::AddRotateKey

HRESULT
AddRotateKey(D3DVALUE
rvTime, D3DRMQUATERNION
*rqQuat);

fff fff,%“f ,’%,,

-

Œ,,,D3DRM_OK Ž”,,,ff ,
,, •’,

Direct3D の

rvTime

rvTime 49 の 0 の の の の rvTime 99

rqQon::DeleteKey

xe "IDirect3DRMAnimation\:\:AddScaleKey"\$

IDirect3DRMAnimation::AddScaleKey
HRESULT AddScaleKey(D3DVALUE rvTime, D3DVALUE rvX, D3DVALUE rvY,
D3DVALUE rvZ);

fff fff, ff f f ,<%0
• CE,,,D3DRM_OK Ž",, の Direct3D
の, Ž ,,,

rvTime

fff fff, ff f f ,<%0
,,,ŽŠ 0,CE“,,,”^,ŽŠ,Ž',,, CE,f ,rvTime,'99,,, rvTime,49,,,f ,ff
f fff,,,,,Š,,,

rvX, rvY, rvZ

ŽCE .
,,ffff,,, Ž•Š,“—,,, •Š,Š, —CEŽ,,,,, 3D•Š ,Ž ,,,
Ž IDirect3DRMAnimation::DeleteKey

xe "IDirect3DRMAnimation\:\:DeleteKey"\$

IDirect3DRMAnimation::DeleteKey
HRESULT DeleteKey(D3DVALUE rvTime);

f ,fff fff,, ,,
• CE,,,D3DRM_OK Ž",,ff ,• ,•,^—, Direct3D
の

rvTime

```
xe "IDirect3DRMAnimation\:\:GetOptions"§
IDirect3DRMAnimation::GetOptions
D3DRMANIMATIONOPTIONS GetOptions();
```

Ž □ □ IDirect3DRMAnimation::SetOptions xe "IDirect3DRMAnimation\:\:SetFrame"§

```
IDirect3DRMAnimation::SetFrame
HRESULT SetFrame(LPDIRECT3DRMFAME lpD3DRMFrame);
```

```
fff□fff,ff□f,□',,□
•   E,,,D3DRM_OK                               Direct3D
    f,•' ,Ž',,,,
```

```
lpD3DRMFrame
fff fff, ',ff f,Ž',• ,ffff
```

```
xe "IDirect3DRMAnimation\:\:SetOptions"§
IDirect3DRMAnimation::SetOptions
HRESULT SetOptions(D3DRMANIMATIONOPTIONS d3drmanimFlags);
```

```
fff□fff
•   E,,,D3DRM_OK Ž",,,ff ,•, •',^—
    ,,,, Direct3D•Žf f,•' ,Ž',,,,
```

```
d3drmanimFlags
fff fff fffffŽ',,D3DRMANIMATIONOPTIONSE',ffff
```

```
Ž   IDirect3DRMAnimation::GetOptionst
xe "IDirect3DRMAnimation\:\:SetTime"§
IDirect3DRMAnimation::SetTime
HRESULT SetTime(D3DVALUE rvTime);
```

の

- **D3DRM_OK**のDirect3D の

rvTi,f□,fff□fff,,,,,Š,,,□

IDirect3DRMAnimationSet

*ffff□fff,□Direct3DRMAnimationffffff,ff□f%
,,,,,IDirect3DRMAnimationSetfff□ffff,ffff,Ž-
,,,,,,□,,fff□ffff,,,,,•Ž,fff□fff,-
,,□□,,,,,Š',,,□,,fffff,□IDirect3DRMAnimationSetfff□f
fff□ffff,ffffff,,□š", ∅ IDirect3DRMAnimation
IDirect3DRMAnimationSetfff ffff ,Ž ,,,*

IDirect3DRMAnimationSetfff ffff,ffff, ^%o,ff f,•—,,,,,,

'%o f f

AddAnimation

DeleteAnimation

Load

Žš

SetTime

*IDirect3DRMAnimationSetfff ffff, ,,,COMfff
IUnknownfff ffff ffff,CE ,, IUnknownfff ffff,Ž,,,,ffff,ff f,,*

AddRef

QueryInterface

Release

,,, IDirect3DRMAnimationSetfff ffff, IDirect3DRMObjectfff ffff,
 ,Z,fff,Œ ,,

AddDestroyCallback

Clone

DeleteDestroyCallback

GetAppData

GetClassName

GetName

SetAppData

SetName

Direct3DRMAnimationSet

IDirect3DRM::CreateAnimationSet

xe "IDirect3DRMAnimationSet\:\:AddAnimation"§

, ffffffff, □□, Ž, DWORD' , , ffff NULL

d3drmLOFlags

f f fffff, Ž, D3DRMLOADOPTIONSĀ,

d3drmLoadTextureProc

D3DRMLOADTEXTURECALLBACKf ffffŠ , , , „, ff fff, •—
 , , fffffff, —, , fffff, „, , , , Ā, , , , , fff f, , NULL, Ž' , , , , , ,

lpArgLTP

D3DRMLOADTEXTURECALLBACK

○ f

lpParentFrame

□ ff□ f, fffff□ , , fff□ f, □ fff□ fffffff, ff□ f, □□ f
 f□ f, N, f□ f, , □

xe "IDirect3DRMAnimationSet\:\:SetTime"§

IDirect3DRMAnimationSet::SetTime

HRESULT SetTime(D3DVALUE rvTime);

fff□ fff, Ā□, ŽŠ, □' , , □

- Ā, , , D3DRM_OK
 ○' , Ž , , ,

Direct3D

rvTime

, , ŽŠ

IDirect3DRMDevice

ffff fff, —ffff, 'Ā' , , , , IDirect3DRMDevicefff ffff, ffff, Ž—
 , , , fffff, IDirect3DRMDevicefff ffff ffff, fffffff, , , Š' , , , , Š—
 , IDirect3DRMDevice , IDirect3DRMDeviceArrayfff ffff
 , Ž , , ,

IDirect3DRMDevicefff ffff ffff, ^%, ff f, •—, , , , , , ,

ffff ffff

GetBufferCount

SetBufferCount

f

GetColorModel

GetDither

SetDither

Init

InTrianglesDrawn

GetViewports

GetWidth

GetWireframeOptions

Update

''

AddUpdateCallback

DeleteUpdateCallback

ff ①

GetQuality

SetQuality

GetShades

SetShades

ffff •Ž

GetTextureQuality

SetTextureQuality

Direct3DRMDevice\:\:ffff, □ IDirect3DRM::CreateDevice

xe "IDirect3DRMDevice\:\:AddUpdateCallback"§

IDirect3DRMDevice::AddUpdateCallback

**HRESULT AddUpdateCallback(D3DRMUPDATECALLBACK
d3drmUpdateProc, LPVOID arg);**

IDirect3DRMDevice::Update

- **D3DRM_OK**
Direct3D の

d3drmUpdateProc

の D3DRMUPDATECALLBACK

arg

の

IDirect3DRMDevice::DeleteUpdateCallback,
IDirect3DRMDevice::Update, D3DRMUPDATECALLBACK

xe "IDirect3DRMDevice\:\:DeleteUpdateCallback"§

IDirect3DRMDevice::DeleteUpdateCallback

HRESULT

**DeleteUpdateCallback(D3DRM
UPDATECALLBACK
d3drmUpdateProc,
LPVOID arg);**

**IDirect3DRMDevice::AddUpdate
Callback**

ff

- *Direct3D* D3DRM_OK
 ①

d3drmUpdateProc

D3DRMUPDATECALLBACK

①

arg

ID

**irect3DRMDevice::AddUpdate
Callback,
IDirect3DRMDevice::Update,
D3DRMUPDATECALL**

BACK

xe "IDirect3DRMDevice\:\:GetBufferCount"§

IDirect3DRMDevice::GetBufferCount

DWORD GetBufferCount();

IDirect3DRMDevice::SetBufferCount

,,Ž,,

•

*ffff, ,, 1,ffff ffff 2,2 fff
f ^%03 4 ,,,*

```

xe "IDirect3DRMDevice\:
\:GetColorModel"§
IDirect3DRMDevice::GetColorMo
del
D3DCOLORMODEL
GetColorModel();

```

```

ffff,ff  fff,Ž“,,,

```

•

Direct3D

RGB

D3DCOLORMODEL

```

xe "IDirect3DRMDevice\:\:GetDirect3DDevice"§

```

```

IDirect3DRMDevice::GetDirect3DDevice

```

```

HRESULT Ge のの

```

- D3DRM_OK Ž”,,,ff ,• ,•,^—
Direct3D Žf f,• Ž ,,,

```

lpD3DDevice

```

```

' f f,ffff fffffff,,ffff, Š%o,,,ffff,ffff

```

```

xe "IDirect3DRMDevice\:
\:GetDither"§...ect3DRMDevice\:
\:GetShades"§

```

```

IDirect3DRMDevice::GetShades

```

```

DWORD GetShades();

```

```

%o, □, Ž“,,□,□,□Ž—,,,,□□,•@,,,□

```

- の

IDirect3DRMDevice::SetShades

xe "IDirect3DRMDevice\:\:GetTextureQuality"§
IDirect3DRMDevice::GetTextureQuality
D3DRMTEXTUREQUALITY GetTextureQuality();

∞∞
∞

RGB

• D3DRMTEXTUREQUALITY :\:GetViewports"§

IDirect3DRMDevice::GetViewports

HRESULT GetViewports(LPDIRECT3DRMVIEWPORTARRAY* lpViewports);

ffff, ff□f□f, Direct3DRMViewportArray

- E,,,D3DRM_OK Ž”,,,ff ,•, •,^—
,,,, Direct3D•Žf f,• ,Ž ,,,,

lpViewports

E,□,,□E,,,,,□—E,Direct3DRMViewportArrayff

xe "IDirect3DRMDevice\:\:GetWidth"§

IDirect3D• •,•,□

xe "IDirect3DRMDevice\:\:GetWireframeOptions"§

IDirect3DRMDevice::GetWireframeOptions

DWORD GetWireframeOptions();

,ffff,fff ff f fffff,Ž“,,

- ^%o,’,fffOR,•,,,

D3DRMWIREFRAME_CULL

∞


```

    ,,,,ffff,~,•Ž',,,    “,•Ž,—,,,ffff, ,,, ,•Ž,,,,,
Ž   IDirect3DRMDevice::GetQuality
xe "IDirect3DRMDevice\:\SetShades"§
IDirect3DRMDevice::SetShades
HRESULT SetShades(DWORD ulShades);

```

- □☒,,,D3DRM_OK□Ž”,,,ff□, Direct3D の

ulShades

**□□□□□の□□□□□のt3DRMDevice\:
:\SetTextureQuality"§**

```

IDirect3DRMDevice::SetTextureQuality
HRESULT SetTextureQuality(D3DRMTEXTUREQUALITY tqTextureQuality);

```

- ffff,fffff•Ž,□',,,□
D3DRM_OK Ž”,,,ff ,•, •',^—
,,,, Direct3D•Žf f,•' ,Ž ,,,,

```

tqTextureQuality
D3DRMTEXTUREQUALITY—☒,fff,,, ffff,D3DRMTEXTURE,,,
Ž   IDirect3DRMDevice::GetTextureQuality
xe "IDirect3DRMDevice\:\Update"§
IDirect3DRMDevice::Update
HRESULT Update();

```

- fffffff,ffffff,,,ff f,• ,,, ffff ffff,f ff fŠ ,'<,,
□☒,,,D3DRM_OK□Ž”,, Direct3D
の

```

D3DRMUPDATECALLBACK
IDirect3DRMDevice::AddUpdateCallback

IDirect3DRMDevice::AddUpdateCallback,
D3DRMUPDATECALLBACK

IDirect3DRMFace

```

IDirect3DRMFace の IDirect3DRMFace の IDirect3DRMFace
IDirect3DRMFaceArray

IDirect3DorRGB

ffff

GetMaterial

SetMaterial

ffff

GetTexture

GetTextureCoordinateIndex

GetTextureCoordinates

GetTextureTopology

SetTexture

SetTextureCoordinates

SetTextureTopology

”“,-

AddVertex

AddVertexAndNormalIndexed

GetNormal

GetVertex

GetVertexCount

GetVertexIndex

GetVertices

IDirect3DRMFacefff ffff, ,,,COMfff ffff,“—
 IUnknownfff ffff ffff,Œ , IUnknown
 ,,,ffff,ff f,,
 AddRef

QueryInterface

Release

,,, IDirect3DRMFace fff fff, IDirect3DRMObject の
AddDestroyCallback

Clone

Direct3DRMFace ffffff, IDirect3DRM::CreateFace ffff, CE, ,,,, Ž“ ,,,

```
xe "IDirect3DRMFace\:\:AddVertex"$  
IDirect3DRMFace::AddVertex  
HRESULT AddVertex(D3DVALUE x, D3DVALUE y, D3DVALUE z);
```

Direct3DRMFace f f

- CE,,,D3DRM_OK Ž”,,,ff ,*, *’^—
,,,, Direct3D•Žf f,•’ ,Ž ,,,,

x, y, z

,,“^”,x y z •

```
xe "IDirect3DRMFace\:\:AddVertexAndNormalIndexed"$
```

IDirect3DRMFace::AddVertexAndNormalIndexed

HRESULT AddVertexAndNormalIndexed(DWORD vertex, DWORD normal);

”,- ,Direct3DRMFaceffffff,%o,, ,,, ffff□fff,Š,,’“,ffffff,-
 □,ffffff,Ž—,□-□“ - Direct3DRMMeshBuilder 1

• □ ex, normal

’%o,,’“,-□,ffffff□

```
xe "IDirect3DRMFace\:\GetColor"§
IDirect3DRMFace::GetColor
D3DCOLOR GetColor();
```

Direct3DRMFace の

•

```
IDirect3DRMFace::SetColor
xe "IDirect3DRMFace\:\GetMaterial"§
IDirect3DRMFace::GetMaterial
HRESULT GetMaterial(LPDIRECT3DRMMATERIAL* lpMaterial);
```

Direct3DRMFaceffffff,fffff,Ž“,,

• D3DRM_OK
 Direct3D の

lpMaterial

Direct3DRMMaterial の

```
IDirect3DRMFace::SetMaterial
xe "IDirect3DRMFace\:\GetNormal"§
IDirect3DRMFace::GetNormal
HRESULT GetNormal(D3DVECTOR *lpNormal);
```

Direct3DRMFace

. D3DRM_OK の Direct3D の

lpNormal

D3DVECTOR

xe "IDirect3DRMFace\:\:GetTexture"§

IDirect3DRMFace::GetTexture

HRESULT GetTexture(LPDIRECT3DRMTEXTURE* lpTexture);

Direct3DRMFace

- D3DRM_OK
Direct3D の

lpTexture

の

IDirect3DRMFace::SetTexture

xe "IDirect3DRMFace\:\:GetTextureCoordinateIndex"§

IDirect3DRMFace::GetTextureCoordinateIndex

int Get

TextureCoordinateIndex(DWORD
dwIndex);

–
,,,’“,ffffff,fffff •,Ž“,,
,,ffffff, dwIndexfff f,Ž’,
,,ffffff,‘%o,,
• ffffff,*,

dwIndex

–,,,’“,ffffff

xe "IDirect3DRMFace\:

```

\\GetTextureCoordinates"§
IDirect3DRMFace::GetTexture
Coordinates
HRESULT
GetTextureCoordinates(DWOR
D index, D3DVALUE *lpU,
    D3DVALUE *lpV);

```

```

Direct3DRMFaceffffff',",fff
ff ,Ž",,

```

-

```

œ,,D3DRM_OK Ž",,ff ,•,
•,^—
,,,, Direct3D•Žf f,•' ,Ž ,,,,

```

```

index
"",ffffff
lpU, lpV
"",fffff ,Ž,Ž,•

```

```

xe "IDirect3DRMFace\\:
\\GetTextureTopology"§
IDirect3DRMFace::GetTexture
Topology
HRESULT
GetTextureTopology(BOOL
*lpU, BOOL *lpV);

```

```

Direct3DRMFaceffffff,fffff^
',Ž",,

```

-

D3DRM_OK

Direct3D の

lpU, lpV

IDirect3DRMFace::SetTextureTo
pology

xe "IDirect3DRMFace\
\:GetVertex"§

IDirect3DRMFace::GetVertex
HRESULT GetVertex(DWORD
index, D3DVECTOR *lpPosition,
D3DVEC

TOR *lpNormal);

Direct3DRMFace

- D3DRM_OK
Direct3D

f, ,Ž ,,,

index

“,ffffff

lpPosition, lpNormal

“,^,- ,Ž,Ž,D3DVECTOR “

xe "IDirect3DRMFace\
\:GetVertexCount"§

IDirect3DRMFace::GetVertexCou
nt

red, green, blue

— , —‘

xe "IDirect3DRMFace\:\SetMaterial"§

IDirect3DRMFace::SetMaterial

HRESULT SetMaterial(LPDIRECT3DRMMATERIAL lpD3DRMMaterial);

Direct3DRMFaceffffff,ff

• □☒,,D3DRM_OK□Ž”,,,ff□,f□

Ž□□IDirect3DRMFace::GetMaterial

xe "IDirect3DRMFace\:\SetTexture"§

IDirect3DRMFace::SetTexture

HRESULT SetTexture(LPDIRECT3DRMTEXTURE lpD3DRMTexture);

Direct3DRMFaceffffff,ffffff,fff,,

• □☒,,D3DRM_OK□Ž”

☺

Direct3D

☺

lpD3DRMTextureCoordinates"§

IDirect3DRMFace::SetTextureCoordinates

HRESULT SetTextureCoordinates(DWORD vertex, D3DVALUE u, D3DVALUE v);

Direct3DRMFaceffffff',',",ffffff□,□',,,□

• D3DRM_OK

Direct3D•Žf f,•' ,Ž ,,,,

vertex

' ,',",ffffff ,,,, -,ŽŠ☒,, "0 1 2,,,

u, v

Ž' ,,,',Š,“,ffffff •

xe "IDirect3DRMFace\:\SetTextureTopology"§

IDirect3DRMFace::SetTextureTopology

HRESULT SetTextureTopology(BOOL cyIU, BOOL cyIV);

Direct3DRMFaceffffff,ffffff^', ',,

• □☒,,D3DRM_OK□Ž”,,,ff□,•,□',^—

Direct3D ☺

fff
GetSceneFogColor

GetSr

SetSceneFogEnable

SetSceneFogMode

SetSceneFogParams

§'
AddChild

DeleteChild

GetChildren

GetParent

GetScene

☒
AddLight

DeleteLight

GetLights

Load

GetMaterialMode

SetMaterialMode

~, ~"

AddMoveCallback

AddRotation

AddScale

AddTranslation

DeleteMoveCallback

GetOrientation

GetPosition

GetRotation

GetVelocity

LookAt

Move

SetOrientation

SetPosition

SetRotation

SetVelocity

•—

GetSortMode

GetZbufferMode

SetSortMode

SetZbufferMode

GetTexture

GetTexturm

Transform

ffff □ *ffff*
AddVisual

DeleteVisual

GetVisuals

IDirect3DRMFrame*ffff* □ *ffff*, □, , , , COM*ffff* □ *ffff*, \-
□ IUnknown*ffff* □ *ffff* □ *ffff*, @ □, □ IUnknown*ffff* □ *ffff*, ž, , , , *ffff*, *ff* □ *f*, , □
AddRef

QueryInterface

Release

IDirect3DRMFrame	IDirect3DRMObject
∅	
AddDestroyCallback	
Clone	
DeleteDestroyCallback	
GetAppData	
GetClassName	
GetName	
SetAppData	
SetName	

Direct3DRMFrame □□
IDirect3DRM::CreateFrame *ffff*, **CE**, □, **DIRECT3DRMFRAME** *lpD3DRMFrameChild*);

ff □ **fŠ** ' **Žff** □ **f** , '%0, , □

• □CE, , , D3DRM_OK □ Ž " , , , , ff □ , • , □ • ' , ^ - , , , , □ □ Direct3D • Ž f □ f , • ' □ , Ž □ , , , , □

lpD3DRMFrameChild
Žff □ *f* , , , ' % , , *Direct3DRMFrame* *ffff* □
Žff □ *f* , , , ' % , , *ff* □ *f* , , , , □ , Ž , , , , □ □
 0

AddChild
IDirect3DRMFrame::GetTransform::GetTransform *ffff*, **CE** , *ffff* , Š
 —, Ž " , , , , , , , , , ff f , '%0, , CE , Š , • , , , ,

xe "IDirect3DRMFrame\:\:AddLight"§
IDirect3DRMFrame::AddLight
HRESULT AddLight(**LPDIRECT3DRMLIGHT** *lpD3DRMLight*);
f f , CE , '%0, ,

xe "IDirect3DRMFrame\:\:AddMoveCallback"§

IDirect3DRMFrame::AddMoveCallback
HRESULT AddMoveCallback(**D3DRMFRAMEMOVECALLBACK** *d3drmFMC*, **VOID** *
lpArg);
 "• , ^ " □ - , , , , f □ f f f f Š □ , ' % , , , □
 • **D3DRM_OK**
 , , **Direct3D** • Ž f f , • ' , Ž , , , ,

d3drmFMC
ffff fff ' < , **D3DRMFRAMEMOVECALLBACK** *f ffff* Š
lpArg
f ffff Š , " , , , *ffff fff* ' < , *f f*
 Ž **IDirect3DRMFrame::Move**,
IDirect3DRMFrame::DeleteMoveCallback
 xe "IDirect3DRMFrame\:\:AddRotation"§
IDirect3DRMFrame::AddRotation
HRESULT AddRotation(**D3DRMCOMBINETYPE** *rectCombine*, **D3DVALUE**
rvX,
D3DVALUE *rvY*, **D3DVALUE** *rvZ*, **D3DVALUE** *rvTheta*);

• `D3DRM_OK` `Ž`, „`ff`“, „`•`“, „`•`“, „`^`—
 „`„`“, `Direct3D`•`Ž` `f`, „`•`“, „`Ž`“, „`„`“, „`„`“

lpD3DRMVisual

`„`„, `Direct3DRMVisual` `fffff`, `Ž`, „`•`“, `ffff`„

```
xe "IDirect3DRMFrame\:\:GetChildren"$
IDirect3DRMFrame::GetChildren
HRESULT GetChildren(LPDIRECT3DRMFRAMEARRAY* lpChildren);
```

`Direct3DRMFrameArray` の

• `„`„, `D3DRM_OK` `Ž` „`ff`“, „`•`“, „`•`“, „`^`—
 „`„`“, `Direct3D`•`Ž` `f`, „`•`“, „`Ž`“, „`„`“, „`„`“

`lpChildren`

`„`„, „`„`“, `„`„, „`„`“, —`„`„, `Direct3DRMFrameArray` `ffff`, „`ffff`“, `ffff`

`Ž` `Direct3DRMFrameArray`, `Š`‘

```
xe "IDirect3DRMFrame\:\:GetColor"$
```

`IDirect3DRMFrame::GetColor`

`D3DCOLOR` `GetColor`();

"IDirect3DRMFrame\:\:GetLights"\$

`IDirect3DRMFrame::GetLights`

`HRESULT` `GetLights`(`LPDIRECT3DRMLIGHTARRAY* lpLights`);

`Direct3DRMLightArray` `fffff`, `„`„, `ff`„`„`, `„`„, `„`„, „`„`“, „`„`“, „`„`“

• `D3DRM_OK`
`Direct3D`•`Ž` `f`, „`•`“, „`Ž`“, „`„`“, „`„`“

`lpLights`

`„`„, „`„`“, `„`„, „`„`“, —`„`„, `Direct3DRMLightArray` `ffff`, „`ffff`“, `ffff`

`Ž` `IDirect3DRMLightArray`

```
xe "IDirect3DRMFrame\:\:GetMaterialMode"$
```

`IDirect3DRMFrame::GetMaterialMode`

`D3DRMMATERIALMODE` `GetMaterialMode`();

`f`„`fffff`„`f`, `Ž`

• ④ D3DRMMAentation"§

IDirect3DRMFrame::GetOrientation

HRESULT GetOrientation(LPDIRECT3DRMFRAME lpRef, LPD3DVECTOR lprvDir, LPD3DVECTOR lprvUp);

ff□f,☉,,ž",,,□,,•☉,□ž',,,ž□ff□f,,^^',,ž",,,□

- □☉,, D3DRM_OK

Direct3D の

lpRef

,Direct3DRMFrameffffff,ž,•

lprvDir, lprvUp

ff f,zž,yž,•☉,ž,ž,D3DVECTOR “

ž IDirect3DRMFrame::SetOrientation

xe "IDirect3DRMFrame\:\:GetParent"§

IDirect3DRMFrame::GetParent

HRESULT GetParent(LPDIRECT3DRMFRAME* lpParent);

☉ ,ff f, ff f,ž“,,,

- ☉,,D3DRM_OK

•,^—,.,.,.,

Direct3D•žf f,•' ,ž ,.,.,

lpParent □',,,□

xe "IDirect3DRMFrame\:\:GetPosition"§

IDirect3DRMFrame::GetPosition

HRESULT GetPosition(LPDIRECT3DRMFRAME lpRef, LPD3DVECTOR lprvPos);

ž',,,ž□ff□f,,□ff□f,^^',ž",,,□,, ののの

lprvPos ,,,

- ☉,,D3DRM_OK ž”,,,ff ,•, •,^—,.,.,.,

Direct3D•žf f,•' ,ž ,.,.,

lpRef

ž ,,—,.,.,Direct3DRMFrameffffff,ž,•

lprvPos

ff f,^,ž,ž,D3DVECTOR “

ž□□IDirect3DRMFrame::SetPosition

xe "IDirect3DRMFrame\:\:GetRotation"§

IDirect3DRMFrame::GetRotation
 HRESULT GetRotation(LPDIRECT3DRMFRAME lpRef, LPD3DVECTOR
 lprvAxis,

LPD3DVALUE lprvTheta);

f f □ *f*, % "Ž", □ % "Ž",

- D3DRM_OK
Direct3D の

lpRef

Direct3DRMFrame

lprvAxis

D3DVECTOR

lprvTheta

の

IDirect3DRMFrame::SetRotation, Transformations

xe "IDirect3DRMFrame\:\:GetScene"§

IDirect3DRMFrame::GetScene

HRESULT GetScene(LPDIRECT3DRMFRAME lpRoot);*

- D3DRM_OK
 ,,,, Direct3D Ž f f, Ž
 ,,,,

lpRoot

*f f, ^ f f, Ž, Direct3DRM
 Frame f f f f, Ž, Ž, f f f f, f f f f*

*xe "IDirect3DRMFrame\
 \:GetSceneBackground"§*

IDirect3DRMFrame::GetSceneBackground
D3DCOLOR
GetSceneBackground();

f f, "Ā ,Ž", ,
• ,;

**xe "IDirect3DRMFrame\
:\GetSceneBackgroundDepth"§**
IDirect3DRMFrame::GetSceneBackgroundDepth

HRESULT
GetSceneBackgroundDepth(

LPDIRECTDRAWSURFACE *
lpDDSsurface);

f f,Ā , "Ā "ffff,Ž", ,
•

Ā,,,D3DRM_OK Ž",,,ff ,
, , , ^—
,,,, Direct3D•Žf f,• ,Ž
""

lpDDSsurface
Ā , "Ā "ffff,Ž,DirectDraw
f fff,ffff,"ffff,ffff
Ž IDirect3DRMFrame::SetSceneBackgroundDepth
**xe "IDirect3DRMFrame\
:\GetSceneFogColor"§**
IDirect3DRMFrame::GetSceneFogColor
D3DCOLOR
GetSceneFogColor();

f f,fff, ,Ž", ,
• fff, ,;

**xe "IDirect3DRMFrame\
:\GetSceneFogEnable"§**

```

IDirect3DRMFrame::GetSceneFogEnable
BOOL GetSceneFogEnable();

```

```

„f f, Ć fff,—Ć,,,,,,*,
• fff,—
Ć,,,,TRUE ,,,,,,FALSE,*,

```

```

xe "IDirect3DRMFrame\:
\:GetSceneFogMode"§
IDirect3DRMFrame::GetSceneFogMode
D3DRMFOGMODE
GetSceneFogMode();

```

```

„f f,Ć ,fff f f,*,
•
Ć ,fff f f,Ž,D3DRMFOG
MODE—Ć,fff,,,,*,

```

```

xe "IDirect3DRMFrame\:
\:GetSceneFogParams"§
IDirect3DRMFrame::GetSceneFogParams
HRESULT
GetSceneFogParams(D3DVALUE * lprvStart, D3DVALUE * lprvEnd,
    D3DVALUE * lprvDensity);

```

```

„f f,Ć ,fff fff f,Ž“„,
•
Ć,,,D3DRM_OK Ž”„,ff ,•
, •,^—
„,,, Direct3D•Žf f,•’ ,Ž
””

```

```

lprvStart, lprvEnd, lprvDensity
fff,Ž“ “ -“,’Ž,Ž,•

```

```

xe "IDirect3DRMFrame\:

```

```
\:GetSortMode"§
IDirect3DRMFrame::GetSortM
ode
D3DRMSORTMODE
GetSortMode();
```

```
Žff f, —,,,,Ž,,•—
f f,Ž“,,,
• •—
f f,Ž,D3DRMSORTMODE
—(E,fff,•,
```

```
Ž IDirect3DRMFrame::SetS
ortMode
xe "IDirect3DRMFrame\
\:GetTexture"§
IDirect3DRMFrame::GetTextur
e
HRESULT
GetTexture(LPDIRECT3DRMT
EXTURE* lpTexture);
```

```
Ž',,,ff f,fffff,Ž“,,,
•
E,,D3DRM_OK Ž",,,ff ,
•, •',^—
,,,, Direct3D•Žf f,•' ,Ž
,,,,
```

```
lpTexture
ff f,fffff,Ž,Dir
```


ct3DRMTexture

IDirect3DRMFrame::SetTexture

xe "IDirect3DRMFrame\:\:GetTextureTopology"§

IDirect3DRMFrame::GetTextureTopology

HRESULT GetTextureTopology(BOOL * lpbWrap_u, BOOL * lpbWrap_v);

Ž',,,ff f ,ffffff,fff,,,,, ffff
f,^“ ,Ž“,,

- **D3DRM_OK**
Direct3D の

lpbWrap_u, lpbWrap_v

u

v の TRUE

Ž **IDirect3DRMFrame::SetTextureTopology**

xe "**IDirect3DRMFrame\:\:GetTransform**"§

IDirect3DRMFrame::GetTransform

HRESULT
GetTransform(D3DRMMATRIX4
TransformMatrix);

4 4—,fff —
,,ff f,f ff•Š,Ž“,,

- **CE,,D3DRM_OK Ž”,,ff**

∅ Direct3D の

rmMatrix

D3DRMMATRIX4D の

3D

xe "IDirect3DRMFrameLPDIRECT3DRMFRAME lpRef, LPD3DVECTOR lprvVel,

BOOL fRotVel);

Ž',,,ž□f□f, \, ,□f□f, \'\',ž",,□

- □Œ,,D3DRM_OK□Ž",,,ff□,•,□•',^—,,,,, Direct3D の

lpRef

,,,—,,,,Direct3DRMFramefffff,Ž,•

lprvVel

ff f,“,ž,ž,D3DVECTOR “,ffff

fRotVel

^žŒ,“ž“,, ffffff,%o“““ —

“,,,,,,ž,fff ,fff f,TRUE,ž',,fffff,%o“““ ,Œž,š,,, Ž IDirect3DRMFrame::SetVelocity

xe "IDirect3DRMFrame\:\:GetVisuals"§

IDirect3DRMFrame::GetVisuals

HRESULT GetVisuals(LPDIRECT3DRMVISUALARRAY* lpVisuals);

のの

- D3DRM_OK
Direct3D の

lpVisuals

Direct3DRMVisualArray

xe "IDirect3DRMFrame\:\:GetZbufferMode"§

IDirect3DRMFrame::GetZbufferMode

D3DRMZBUFFERMODE GetZbufferMode();

Z

Z

• D3DRMZBUFFT3DRMFrame\: \:InverseTransform"§

IDirect3DRMFrame::InverseTransform

HRESULT InverseTransform(D3DVECTOR *lprvDst, D3DVECTOR *lprvSrc);

xe "IDirect3DRMFrame\:\:LPDIRECT3DRMFRAME lpRef,

D3DRMFRAMECONSTRAINT rfcConstraint);

Ž',,,,Ž□ff□f,"Š,,□ff□f,-•ff□f,☉,□-,,,,□☉,%",fff,,□
 • □☉,,,D3DRM_OK□Ž",,ff□,•□,^—,□□Direct3D•Žf□f,•□,Ž□,□□

lpTarget, lpRef
 -•,,,Ž□,,-,,,Direct3DRMFramefffff,Ž,•□□
 rfcConstraint

D3DRMFRAMECONSTRAINT

xe "IDirect3DRMFrame\:\:Move"§

IDirect3DRMFrame::Move

HRESULT Move(D3DVALUE delta);

—,,,Š',,,,ff f,%o,"“,"—,,
 • ☉,,,D3DRM_OK Ž",,ff,•, •,^—
 ,,,, Direct3D•Žf f,•' ,Ž ,,,,

delta

• ,,"%o",—

xe "IDirect3DRMFrame\:\:SetColor"§

IDirect3DRMFrame::SetColor

HRESULT SetColor(D3DCOLOR rcColor);

ff f, , ',, D3DRMMATERIALMODERIALMODE

—(☉,D3DRMMATERIAL_FROMFRAME ☉ ☉ ☉
 ,Ž—,.,,

• □☉,,,D3DRM_OK□Ž Direct3D
 ☉

rcColor

☉

IDirect3DRMFrame::GetColor, IDirect3DRMFrame::SetMaterialMode

xe "IDirect3DRMFrame\:\:SetColorRGB"§

IDirect3DRMFrame::SetColorRGB

**HRESULT SetColorRGB(D3DVALUE rvRed, D3DVALUE rvGreen,
D3DVALUE rvBlue);**

**D3DRMMATERIALMODE
D3DRMMATERIAL_FROMFRAME**

• **CE,,,D3DRM_OK Ž”,,,ff □,•,□,•’,^—
,,,,,□□rect3D•Žf □f,•’ □,Ž□,,,,□**

rvRed, rvGreen, rvBlue

ff□f,□,□□š□—\, 0,,1,“^,,,,,,□
Ž□□IDirect3DRMFrame::SetMaterialMode
xe "IDirect3DRMFrame\:\:SetMaterialMode"§
IDirect3DRMFrame::SetMaterialMode
HRESULT SetMaterialMode(D3DRMMATERIALMODE rmmMode);

∅∅

• **CE,,,D3DRM_OK Ž”,,,ff ,•, •’,^—
,,,,, Direct3D•Žf f,•’ ,Ž ,,,,**

rmmMode
D3DRMMATERIALMODE—(Cfff ,,,,
Ž IDirect3DRMFrame::GetMaterialMode
xe "IDirect3DRMFrame\:\:SetOrientation"§
IDirect3DRMFrame::SetOrientation
HRESULT SetOrientation(LPDIRECT3DRMFRAME lpRef, D3DVALUE
rvDx,
D3DVALUE rvDy, D3DVALUE rvDz, D3DVALUE rvUx, D3DVALUE
rvUy,
D3DVALUE rvUz);

ff□f,• z [rvDx, rvDy, rvDz] y [rvUx,
rvUy, rvUz]

- `HRESULT D3DRM_OK`
`Dx, rvDy, rvDz`

`ff f, ,zŽ`

`rvUx, rvUy, rvUz`

`ff f, ,yŽ`

`ffff, fff, fffff[0, 0, 1], fffff[0, 1, 0], Ž,,, [rvUx, rvUy, rvUz], [rvDx, rvDy, rvDz], •, D3DRMERR_BADVALUE f f', •, , , ^Š, , , [rvUx, rvUy, rvUz] ffff, [rvDx, rvDy, rvDz]`

`IDirect3DRMFrame::GetOrientation`

`xe "IDirect3DRMFrame\:\:SetPosition"§`

`IDirect3DRMFrame::SetPosition`

`HRESULT SetPosition(LPDIRECT3DRMFRAME lpRef, D3DVALUE rvX, D3DVALUE rvY, D3DVALUE rvZ);`

`Ž ff f,, ff f,“^”, ’, ff f, Ž ff f,,<—,[rvX, rvY, rvZ], ’, ”” ff f,[0, 0, 0], ^, ””`

- `•, D3DRM_OK Ž”,, ff ,•, •’, ^—, Direct3D`
`Ⓣ’ ,Ž ,,,`

`lpRef`

`Ž ,,—,, Direct3DRMFrame ffffff, Ž, • ,ffff`

`rvX, rvY, rvZ`

`Ⓣ`

`IDirect3DRMFrame::GetPosition`

`xe "IDirect3DRMFrame\:\:SetRotation"§`

`IDirect3DRMFrame::SetRotation`

`HRESULT SetRotation(LPDIRECT3DRMFRAME lpRef, D3DVALUE rvX, D3DVALUE rvY,`

`D3DVALUE rvZ, D3DVALUE rvTheta);`

f f,"E f f ,Ž',,
 • ,,D3DRM_OK Ž",,,ff ,*, *,'^—
 ,,,, Direct3D•Žf f,•' ,Ž ,,,,

lpTexture
 ,, Direct3DRMTexture

のののの

xe "IDirect3DHRRESULT SetSceneBackgroundRGB(D3DVALUE rvRed, D3DVALUE rvGreen,

D3DVALUE rvBlue);

f□f,"E□,□',,□
 • □E,,,D3DRM_OK□Ž" , , , ff□,*, □,'^— Direct3D の

rvRed, rvGreen, rvBlue

の

xe "IDirect3DRMFrame\:\:SetSceneFogColor"
 IDirect3DRMFrame::SetSceneFogColor
 HRESULT SetSceneFogColor(D3DCOLOR rcColor);

f f,fff, , ',,
 • E,,,D3DRM_OK Ž",,,ff ,*, *,'^—
 ,,,, Direct3D•Žf f,•' ,Ž ,,,,

rcColor
 の

xe "IDirect3DRMFrame\:\:SetSceneFogEnable"§
 IDirect3DRMFrame::SetSceneFogEnable

HRESULT SetSceneFo.□E,,,D3DRM_OK□Ž" , , , ff□,*, □,'^— —,,,,□□Direct3D•Žf□f,•' □,Ž□,,,,□

Zffff□f□f,□' Z

• □œ,,,D3DRM_OK□ž”,,,ff□,•,□•,^—
,,,,□□Direct3D•žf□f,•'□,ž□,,□

d3drmZBM

Zffff□f□f,ž„D3DRMZBUFFERMODE
—(œ,fff,,,□ffff',D3DRMZBUFFER_FROMPARENT,,,□
ž□□IDirect3DRMFrame::GetZbufferMode

xe "IDirect3DRMFrame\:\:Transform"§

IDirect3DRMFrame::Transform

HRESULT Transform(D3DVECTOR *lpd3dVDst, D3DVECTOR *lpd3dVSrc);

lpd3dVSrc □□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□ **lpd3dVDst** □□□□□□□□
□□•,□□Direct3D•žf□f,•'□,ž□,,□

lpd3dVDst

•š□—,œ%„ž,ž,D3DVECTOR□`□
lpd3dVSrc
•š□—,,,D3DVECTOR□`□
ž□□IDirect3DRMFrame::InverseTransform, 3D•š
IDirect3DRMLight
ffff□fff, □œ IDirect3DRMLight のの
IDirect3DRMLight □, š”,,,š—
, IDirect3DRMLight,IDirect3DRMLightArrayfff ffff ,ž”,,,
IDirect3DRMLightfff ffff ffff, ^%„ff f,•—,,,,,,
œ
GetConstantAttenuation

GetLinearAttenuation

GetQuadraticAttenuation

SetConstantAttenuation

SetLinearAttenuation

SetQuadraticAttenuation

GetColor

SetColor

SetColorRGB

—E”[^]
GetEnableFrame

SetEnableFrame

GetType

SetPenumbra

SetUmbra

IDirect3DRMLight *fff* *fff*, *□*, ..., COM *fff* *fff*, “—
 □ IUnknown *fff* *fff* *fff* *fff*, *⊕* *□*, *□* IUnknown *fff* *fff* *fff*, *ž*, ..., *fff*, *ff* *□* *f*, *□*
 AddRef

QueryInterface

Release

IDirect3DRMLight *fff* *fff*, IDirect3DRMObject *fff* *fff*, *ž*, *fff*, *⊕*
 ”
 AddDestroyCallback

Clone

DeleteDestroyCallback

GetAppData

GetClassName

GetName

SetAppData

SetName

Direct3DRMLight □□□□□□□□
IDirect3DRM::CreateLight □□□
IDirect3DRMLight::GetColor
D3DCOLOR GetColor();

☒□,Direct3DRMLightfffff,□,ž\,,□
 • □,•,□

ž□□IDirect3DRMLight::SetColor
 xe "IDirect3DRMLight\:\.GetConstantAttenuation"\$
 IDirect3DRMLight::GetConstantAttenuation
 D3DVALUE GetConstantAttenuation();

Direct3DRMLight の

• ☒ ' ,•,

☒ ' ,',""—“,☒<“,%o<,—,, ,,,, ☒ ' ,2,,, ☒,<“,”•,%o,,

ž IDirect3DRMLight::SetConstantAttenuation

xe "IDirect3DRMLight\:\.GetEnableFrame"\$

IDirect3DRMLight::GetEnableFrame

HRESULT GetEnableFrame(LPDIRECT3DRMFRAME * lpEnableFrame);

☒☒, ' ,,,,—☒,ff f,ž“,,


```
IDirect3DRMLight::GetType
D3DRMLIGHTTYPE GetType();
```

- D3DRMLIGHTTYPE

```
IDirect3DRMLight::SetType
xe "IDirect3DRMLight\:\:GetUmbra"$
IDirect3DRMLight::GetUmbra
D3DVALUE GetUmbra();
```

```
Direct3DRMLightfffff,-,Š,Ž,,
• -,Š,•,
```

```
Ž IDirect3DRMLight::SetUmbra
xe "IDirect3DRMLight\:\:SetColor"$
IDirect3DRMLight::SetColor
HRESULT SetColor(D3DCOLOR rcColor);
```

```
Ž
• Ć,,D3DRM_OK Ž",,,ff ,•, •',^—,,,,, Direct3D
,,,,
"
```

```
rcColor
ĆĆ, ,,
Ž □ □ IDirect3DRMLight::GetColor
xe "IDirect3DRMLight\:\:SetColorRGB"$
```

IDirect3DRMLight::SetCovBlue);

```
Ž',,,ĆĆĆ, □, R, B, G,,,,', □', □
• □Ć,, ,D3DRM_OK□Ž",,,ff□,•,□•',^—,,,,,□□Direct3D•žf□f,•'□,ž□,,,,□
```

```
rvRed, rvGreen, rvBlue
ĆĆ, □, □□
```

```
xe "IDirect3DRMLight\:\:SetConstantAttenuation"$
```


QueryInterface

Release

• `CE,,,D3DRM_OK` `Ž` `ff` `,` `,` `,` `^` `—`
`,,,,,` `Direct3D` `Žf` `f,` `Ž` `,` `,` `,` `,` `,`

lpr, lpg, lpb

`ffff,` `CE,` `,` `,` `,` `Ž,` `,` `—` `,` `—` `,` `Ž,` `Ž,` `ffff` `,`

```

Ž IIDirect3DRMMaterial::SetEmissive
xe "IDirect3DRMMaterial\:\:GetPower"$
IDirect3DRMMaterial::GetPower
D3DVALUE GetPower();

```

`Ž'` `,` `,` `,` `fffff,` `"ŽŽ` `,` `<` `"` `Ž"` `,` `,` `,` `,` `,`
• `"ŽŽ` `,` `<` `"` `Ž'` `,` `,` `,` `,` `,`

`Ž` `Direct3DRMMaterial::SetPower`

```

xe "IDirect3DRMMaterial\:\:GetSpecular"$
IDirect3DRMMaterial::GetSpecular
HRESULT GetSpecular(D3DVALUE *lpr, D3DVALUE *lpg, D3DVALUE
*lpb);

```

`fffff,` `Žfffff,` `Ž` `,` `,`

• `CE,,,D3DRM_OK` `Ž` `ff` `Direct3D` `Ž`

`lpr, lpg, lpb` `,` `,` `,` `Žfffff,` `—` `,` `—` `,` `Ž,` `Ž,` `fffff`

Ž IIDirect3DRMMaterial::SetEmissive

```

HRESULT SetEmissive(D3DVALUE r, D3DVALUE g, D3DVALUE b);

```

`fffff,` `Ž` `,` `,` `,` `,` `,`
• `Direct3D` `D3DRM_OK` `f,` `,` `Ž` `,` `,` `,` `,` `,`

`r, g, b` `Ž,` `,` `—` `—` `,`

```

Ž IIDirect3DRMMaterial::GetEmissive
xe "IDirect3DRMMaterial\:\:SetPower"$
IDirect3DRMMaterial::SetPower
HRESULT SetPower(D3DVALUE rvPower);

```

Ž

• `CE,,,D3DRM_OK Ž”,,,ff ,•, •’,^—
 ,,,,, □ □ Direct•Žf □ f,•’ □,Ž □,,,, □`

rvPower

□ ,,”ŽŽ □ □

Ž □ □ IDirect3DRMMaterial::GetPower

xe "IDirect3DRMMaterial\:\:SetSpecular"§

IDirect3DRMMaterial::SetSpecular

HRESULT SetSpecular (D3DVALUE r, D3DVALUE g, D3DVALUE b);

○ f, , ’,,

• `CE,,,D3DRM_OK Ž”,,,ff ,•, •’,^—
 ,,,,, Direct3D•Žf f,•’ ,Ž ,,,,`

r, g, b

”Žffff, , — ,Š—‘

Ž IDirect3DRMMaterial::GetSpecular

IDirect3DRMMesh

ffff □ fff, ○

IDirect3DRMMeshfff □ ffff, ffff, Ž—

., □ , , ffff, IDirect3DRMMeshfff □ ffff □ ffff, ffffffff, , , □ Š” , , , , Š—

, □ □ IDirect3DRMMesh, IDirect3DRMMeshBuilder

GetGroupColoupCount

fffff

GetGroupMaterial

SetGroupMaterial

,,`
Scale

Translate

ffffff•ž
GetGroupQuality

SetGroupQuality

fffff□fffff
GetGroupMapping

SetGroupMapping

GetGroupTexture

SetGroupTexture

GetVertices

SetVertices

IDirect3DRMMesh *ffff, ,,,COMfff ffff,“—*
IUnknownfff ffff ffff,Œ ,, IUnknownfff ffff,ž,,,ffff,ff f,
,
AddRef

QueryInterface

Release

,,, □IDirect3DRMMeshfff □fff, IDirect3DRMObjectfff
AddDestroyCallback

Clone

DeleteDestroyCallback

GetAppData

GetClassName

GetName

SetAppData

SetName

Direct3DRMFrame

IDirect3DRM::CreateMesh

xe "IDirect3DRMM unsigned fCount,

unsigned vPerFace, unsigned *fData, D3DRMGROUPINDEX *returnId);

-, □□, ff□f%, □, , ff□f, ž•ž, ž`,, □

- □Œ,,D3DRM_OK

Direct3D

vCount, fCount

ff f,š,,,"-,

vPerFace

,,,-,“, ,”ž, -“,,,” ff f’,-ž,”“, ;^,,,,, ,,fff f,0,ž’,,,,,,,

fData

- D3DRM_OK Ž”,,,ff ,•, •’^—
,,,, Direct3D•Žf f,•’ ,Ž ,,,,

id

ff f,Ž•Ž ,Ž•Ž, IDirect3DRMMesh::AddGroup,Ž,, ,,,,,,,,,,,,,,

D3DRMRENDERQUALITYĈ,•,,—ĈĈ, ’ ,,,,’, ff

value

IDirect3DRMMesh::GetGroupQuality

xe "IDirect3DRMMesh\:\:SetGroupTexture"§

IDirect3DRMMRMTEXTURE value);

Direct3DRMMeshfffff’,ff□f,fffff,Š~•,,□

- □Ĉ,,,D3DRM_OK□Ž”,,,ff□,•,□•’^—,,,,,□□Direct3D•Žf□f,•’□,ž□,,,,□

id

ff□f,ž•ž□,ž•ž, IDirect3DRMMesh::AddGroup,ž,,□□,,,,,,,,,,,,,

,

value

Direct3DRMMesh

IDirect3DRMTexture の

IDirect3DRMMesh::GetGroupTexture

xe "IDirect3DRMMesh\:\:SetVertices"§

IDirect3DRMMesh::SetVertices

HRESULT SetVertices(D3DRMGROUPINDEX id, unsigned index,
unsigned count, D3DRMVERTEX *values);

Ž’,,,ff f,“^, ’,,

- □Ĉ,,,D3DRM_OK□Ž”,,,ff□,•,□•’^—,,,,,□□Direct3D•Ž □ の

id

の の
IDirect3DRMMesh::AddGroup,Ž,, ,,,,,,,,,,,,,,

index

“^, ’,,Ž“Ž,□valuesfff□

count

index

values

GetColorSoace

AddFaces

CreateFace

GetFaceCount

GetFaces

f*□*f
Load

fff
AddMesh

CreateMesh

,'
AddFrame

AddMeshBuilder

ReserveSpace

Save

Scale

SetMaterial

Translate

-□

AddNormal

GenerateNormals

SetNormal

◀

GetPerspective

SetPerspective

•Ž

GetQuality

SetQuality

fffff

GetTextureCoordinates

SetTexture

SetTextureCoordinates

SetTextureTopology

AddVertex

GetVertexColor

GetVertexCount

GetVertices

SetVertex

SetVertexColor

SetVertexColorRGB

IDirect3DRMMeshBuilderfff ffff, ,,,COMfff ffff,“—
 IUnknownfff ffff ffff,Œ ,, IUnknownfff ffff,Ž,,,ffff,ff f,
 ,
 AddRef

QueryInterface

Release

,, IDirect3DRMMeshBuilder
 ,IDirect3DRMObjectfff ffff,,Ž,ffff,Œ ,,
 AddDestroyCallback

Clone

DeleteDestroyCallback

GetAppData

GetClassName

GetName

normalCount

lpNormals

of D3DVECTOR

lpFaceData

normalCount 0

fffff,ff, ff, ffffff 2"- -" -
fffff,Ž, fffff,fff, 0, ',,,,,,

lpD3DRMFaceArray

ffff,Š", IDirect3DRMFaceArrayfff ffff,ffff,ffff

xe "IDirect3DRMMeshBuilder\:\AddFrame"§

IDirect3DRMMeshBuilder::AddFrame

HRESULT AddFrame(LPDIRECT3DRMFRAME lpD3DRMFrame);

Direct3DRMMeshBuilderfffff,%o,

,,□—,□,,,□œff□f;•□,,,Ž□,,,,,□

xe "IDirect3DRMMeshBuilder\:\AddMesh"§

IDirect3DRMMeshBuilder::AddMesh

HRESULT AddMesh(LPDIRECT3DRMMESH lpD3DRMMesh);

ffff,Direct3DRMMeshBuilderfffff,%o,□

• □œ,,D3DRM_OKŽ", , , ff□,•□•', ^—,,,□□Direct3D•Žf□f,•'

lpD3DRMMesh

xe "IDirect3DRMMeshBuilder\:\AddMeshBuilder"§

IDirect3DRMMeshBuilder::AddMeshBuilder

HRESULT AddMeshBuilder(LPDIRECT3DRMMESHBUILDER

lpD3DRMMeshBuild);

Direct3DRMMeshBuilder of Direct3DRMMeshBuilderfffff,%o,,

• œ,,D3DRM_OK Ž",,ff ,•, •', ^—
,,,, Direct3D•Žf f,•' ,Ž ,,,

lpD3DRMMeshBuild

Direct3DRMMeshBuilder ffffffff,ffff ,fffffff,fffff,'%o,,
,, —, ,, ① Direct3DRMMeshBuilder ffffffff,• ,,ž ,,,,,,

xe "IDirect3DRMMeshBuilder\:\:AddNormal"§
IDirect3DRMMeshBuilder::AddNormal

int AddNormal(D3DVALUE x, D3DVALUE y, D3DVALUE z);

–□,Direct3DRMMeshBuilder

- ①

x, y, z

① x y z ①

xe "IDirect3DRMMeshBuilder\:\:AddVertex"§

IDirect3DRMMeshBuilder::AddVertex

int AddVertex(D3DVALUE x, D3DVALUE y, D3DVALUE z);

Direct3DRMMeshBuilder

-

x, y, zr::CreateFace

**HRESULT CreateFace(LPDIRECT3DRMFACE*
lpD3DRMFace);**

"ž,,-,□, □□,□Direct3DRMMeshBuilder ffffffff,'%o,,□

- ①,,D3DRM_OK ž", ① Direct3D
žf f,• ,ž ,,,,

lpD3DRMFace

,,-, ,fff,š",,IDirect3DRMFacefff ffff,ffff,ffff

xe "IDirect3DRMMeshBuilder\:\:CreateMesh"§

Direct3DRMMeshBuilder

- D3DRMCOLORSOURCE の

IIDirect3DRMMeshBuilder
int GetFaceCount();

Direct3DRMMeshBuilder *fffff*, *-*, *□*, *Ž*“, *„*, *□*

- *-*, *□*, *•*, *□*

```
xe "IIDirect3DRMMeshBuilder\:\:GetFaces"§
IIDirect3DRMMeshBuilder::GetFaces
HRESULT GetFaces(LPDIRECT3DRMFACEARRAY*
lpD3DRMFaceArray);
```

Direct3DRMMeshBuilder *fffff*, *-*, *Ž*“, *„*, *□*

- *„*, *„*, D3DRM_OK *Ž*“, *„*, *ff* *„*, *•*, *•*, *^*—
„, *„*, Direct3D *Ž* *f*, *•* *Ž* *„*, *„*

lpD3DRMFaceArray

-, *ffff*, *Ž*, *Ž*, IIDirect3DRMFaceArray *fff* *□* *ffff*, *„*, *ffff*, *f*

```
xe "IIDirect3DRMMeshBuilder\:\:GetPerspective"§
IIDirect3DRMMeshBuilder::GetPerspective
BOOL GetPerspective();
```

Direct3DRMMeshBuilder

- *„*, *„*, TRUE *„*, *„*, FALSE *„*, *„*

xe "IIDirect3DRMMeshBuilder\:\:GetQuality"§

IIDirect3DRMMeshBuilder::GetQuality

D3DRMRENDERQUALITY GetQuality();


```
xe "IDirect3DRMMeshBuilder\:\Save"§
IDirect3DRMMeshBuilder::Save
```

```
HRESULT Save(const char * lpFilename,
```

```
    D3DRMXOFFFORMAT d3drmXOFFFormat,
    D3DRMSAVEOPTIONS ...—
    ,,,, □ □ Direct3D Ž f f, • ' □, Ž □ ,,,, □
```

color

```
—, □ □
```

```
xe "IDirect3DRMMeshBuilder\:\SetColorRGB"§
IDirect3DRMMeshBuilder::SetColorRGB
HRESULT SetColorRGB(D3DVALUE red, D3DVALUE green,
D3DVALUE blue);
```

```
Direct3DRMMeshBuilder ffffff ,,,, —, Ž' ,,, , ' ,,,
• E,,, D3DRM_OK Ž" ,,, ff ,•, •' ^—
    ,,,, Direct3D Ž f f, •' ,Ž ,,,,
```

red, green, blue

```
' , , — ,Š—'
```

```
xe "IDirect3DRMMeshBuilder\:\SetColorSource"§
IDirect3DRMMeshBuilder::SetColorSource
HRESULT SetColorSource(D3DRMCOLORSOURCE source);
```

Direct3DRMMeshBuilder

- D3DRM_OK
Direct3D の

source

D3DRMCOLORSOURCE

```
IDirect3DRMMeshBuilder::GetColorSource
```

```
xe "IDirect3DRMMeshBuilder\:\SetMaterial"§
```


- `D3DRM_OK`

`IDirect3DRMMeshBuilder::GetTextureCoordinates`

```
xe "IDirect3DRMMeshBuilder\:\:SetTextureTopology"$
IDirect3DRMMeshBuilder::SetTextureTopology
HRESULT SetTextureTopology(BOOL cylU, BOOL cylV);
```

Direct3DRMMeshBuilder

- `D3DRM_OK`

cylU, cylV

```
uŽŒ,vŽŒ,^fff uŽŒ,,vŽŒ,%o^,Ž,, ,TRUE ,,,,,,FALSE,Ž',,
```

```
xe "IDirect3DRMMeshBuilder\:\:SetVertex"$
IDirect3DRMMeshBuilder::SetVertex
HRESULT SetVertex(DWORD index, D3DVALUE x, D3DVALUE y,
D3DVALUE z);
```

Direct3DRMMeshBuilderfffff,Ž',,"^, ',,

- `D3DRM_OK`

`IDirect3DRMMeshBuilder::SetVertexColor`

```
IDirect3DRMMeshBuilder::SetVertexColor
HRESULT SetVertexColor(DWORD index, D3DCOLOR color);
```

Direct3DRMMeshBuilderfffff,Ž',,,',",,□,□',,,□

- `D3DRM_OK`

index

of

color

Ž IDirect3DRMMeshBuilder::GetVertexColor

```
xe "IDirect3DRMMeshBuilder\:\:SetVertexColorRGB"$
IDirect3DRMMeshBuilder::SetVertexColorRGB
```

HRESULT SetVertexColorRGB(DWORD index, D3DVALUE red, D3DVALUE green, D3DVALUE blue);

Direct3DRMMeshBuilder, D3DRM_OK, Direct3D の

- D3DRM_OK, Direct3D の

index

red, green, blue

□□□

○ □□□□, Š—‘□

xe "IDirect3DRMMeshBuilder\:\:Translate"§

IDirect3DRMMeshBuilder::Translate

HRESULT Translate(D3DVALUE tx, D3DVALUE ty, D3DVALUE tz);

Direct3DRMMeshBuilder

- D3DRM_OK
Direct3D の

tx, ty, tz

○x y z

IDirect3DRMObject

Direct3DRM

IDirect3DRMObject の

IDirect3DRMObject の

,,,Š—

, IDirect3DRMMaterialfff
ffff ,Ž ,,,

IDirect3DRMObjectfff ffff

ffff, ~%,ff f,•—,,,,,

ffff fff€—,f f

GetAppData

SetAppData

•

Clone

—‘

GetClassName

GetName

SetName

”

AddDestroyCallback

DeleteDestroyCallback

IDirect3DRMObject*fff ffff,*
,,,COMfff ffff;“—
IUnknown*fff ffff ffff,Œ*
„ IUnknownfff ffff,Ž,,,
ffff,ff f,,

AddRef

QueryInterface

Release

Direct3DRMObjectfffff, “
,QueryInterfaceffff,CE, ,,,,
,,Direct3DRMfffff,,,Ž“,,
,,,, ,,,,Direct3DRMfffff, I
Direct3DRMObjectfff ffff
ffff,CE ,,

xe "IDirect3DRMObject\
\:AddDestroyCallback"§

**IDirect3DRMObject::AddDestroyC
allback**

HRESULT
**AddDestroyCallback(D3DRMO
BJECTCALLBACK lpCallback,
LPVOID lpArg);**

fffff,”Š,,,,,CE, ,,Š ,“,,

- D3DRM_OK

Direct3D の

lpCallback

lpArg

fffff,“,,,,,,

のの

xe "IDirect3DRMObject\
\:Clone"§

IDirect3DRMObject::Clone

HRESULT Clone(LPUNKNOWN pUnkOuter, REFIID riid, LPVOID *ppvObj);

- **D3DRM_OK**

pUnkOuter
COM %o<”,%o,,,,,‘ ,fff f
riid
 • „ffffff,Ž•Ž
ppvObj
 ffff, Ę,•,,, ffffff,• ,Š”„fff
 f

xe "IDirect3DRMObject\:
\:DeleteDestroyCallback"§
IDirect3DRMObject::DeleteDestro
yCallback
HRESULT DeleteD

- D3DRM_OKのDirect3D の `fff, -', Ž, NULL, □,, Ž—`
`„ffff□, fff□f, NULL, □□□lpdwSizefff□f,, Ž—, —, fff, □' , , , □`
`Ž□IDirect3DRMObject::SetName`
`xe "IDirect3DRMObject\:\:SetAppData"$`
`IDirect3DRMObject::SetAppData`
`HRESULT SetAppData(DWORD ulData);`

の 32 ' , ,

- `□E,,D3DRM_OK Ž” , , ff , * , * , ^—`
 `, , , , Direct3D Ž f f , * ' , Ž , , ,`

ulData

`fffff, Š” , , f f' ^, f f`

`Ž IDirect3DRMObject::GetAppData`

`xe "IDirect3DRMObject\:\:SetName"$`

`IDirect3DRMObject::SetName`

`HRESULT SetName(const char * lpName);`

`fffff, -', ' , ,`

- `□E,,D3DRM_OK` の
Direct3D の

lpName

`ff, □Š%o, , , , IDirect3DRMShadowfff□ffff, ffff, Ž—`
`„□ffff□fff, IDirect3DRM::CreateShadowfff, □E, □, , , , □, , □Š%, *—`
 `, , , □□Š%, *— , , , □IDirect3DRM::CreateObjectfff, □E, □,`

の IDirect3DRMShadowfff `ffff ffff, fffffff, , , Š” , , , , Š—`
 `, IDirect3DRMShadowfff ffff, Ž , , ,`

IDirect3DRMShadow **Init**

Release

`„□IDirect3DRMShadowfff□fff, □IDirect3DRMObjectfff□ffff, , Ž, ffff, □E, , □`
`AddDestroyCallback`

Clone

DeleteDestroyCallback

GetAppData

GetClassName

GetName

SetAppData

SetName

Direct3DRMShadowfffff, □ IDirect3DRM::CreateShadow

xe "IDirect3DRMShadow\:\:Init"§

IMLight, D3DVALUE px, D3DVALUE py,

D3DVALUE pz, D3DVALUE nx, D3DVALUE ny, D3DVALUE nz) ;

Direct3DRMShadowfffff, □ §%, , □

- Œ,, D3DRM_OK
Direct3D•Ž f ,•' ,Ž ,,,

lpD3DRMVisual

%%%,Direct3DRMVisualfffff

lpD3DRMLight

%%%,E,,E **Direct3DRMLight**

px, py, pz

のの

nx, ny, nz

の の ff□ffff,ffff,Ž—
,,□ffff,CE,ffff”—

,,□,ffff,□IDirect3DRMTextureffff□ffff□ffff,fffff,,□Š”,,
,,Š-,□IDirect3DRMTexture

IDirect3DRMTexture %o,ff f,•—,,,,,,

GetColors

SetColors

ff f

GetDecalOrigin

GetDecalScale

GetDecalSize

GetDecalTransparency

GetDecalTransparentColor rency

SetDecalTransparentColor

ff* *f
GetImage

%
InitFromFile

InitFromResource

InitFromSurface

Changed

GetShades

SetShades

IDirect3DRMTexturefff ffff, ,,,COMfff ffff,“—
IUnknownfff ffff ffff,Œ ,, IUnknownfff ffff,Ž,,,ffff,ff f
”

AddRef

QueryInterface

Release

□□□□IDirect3DRMTextu

Clone

DeleteDestroyCallback

GetAppData

GetClassName

GetName

SetAppData

SetName

Direct3DRMTexture IDirect3DRM::CreateTexture 0 Ž“,,

xe "IDirect3DRMTexture\:\:Changed"§

IDirect3DRMTexture::Changed

HRESULT Changed(BOOL bPixels, **BOOL bPalette**);

□ □

□□□.□,•,□,•,^—

,,,,□□ **Direct3D**•Žf□f,•'□,Ž□,,,,□

bPixels

ffff,•□,,,,,TRUE,□',□

bPalette

ffff,•□,,,,,TRUE,□',□

xe "IDirect3DRMTexture\:\:GetColors"§

IDirect3DRMTexture::GetColors

DWORD GetColors();

fffff,ffffff,Ž,,, ‘ ,Ž“,,

• ,•,

„ffff,□ffffff,□□,Ž,,,ff□f,□□,,,,□,,,,,fffff,□',,,, , ,•, ,,,, ' •,,, ,
, IDirect3DRMTexture::SetColors

,fffff,□,•□,,,E,□IDirect**3DRM**::SetDefaultTextureColors

x"IDirect3DRMTexture\:\:GetImage"\$

IDirect3DRMTexture::GetImage

D3DRMIMAGE * GetImage();

fffff,□□,−,,,ff□f,ffff,•,□

- (E□,ffff,□□,−,,,D3DRMIMAGE□``,`,

xe "IDirect3DRMTexture\:\:GetShades"\$

IDirect3DRMTexture::GetShades

DWORD GetShades();

fffff,ffffff,, ,,,, ,Ž,,,%o%o, ,Ž“,,

- %o%o, ,•,

Ž IDirect3DRMTexture::SetShades

xe "IDirect3DRMTexture\:\:InitFromFile"\$

IDirect3DRMTexture::InitFromFile

HRESULT InitFromFile(const char *filename);

Ž’,,, 0

- □□□□□

D3DRM_ffff,□IDirect3DRM::CreateObject,—

„□□,,,,,,,,,,,,,□

Ž□□IDirect3DRMTexture::InitFromResource,

IDirect3DRMTexture::InitFromSurface

xe "IDirect3DRMTexture\:\:InitFromResource"\$

IDirect3DRMTexture::InitFromResource

HRESULT InitFromResource(HRSRC rs);

Ž’,,,ff f,,Direct3DRMTextureffffff, Š%o,,

- (E,,,D3DRM_OK Ž”,,,ff ,•, •,^—
,,,, Direct3D•Žf f,•’ ,Ž ,,,,

rs

IDirect3DRMTexture::InitFromFile,

IDirect3DRMTexture::InitFromSurface

ff f,ffff,Ž’,,

URFACE lpDDS);

```

Ž ,,,DirectDrawf fff,f f,Ž—,,fffff, Š%oo,,
• E,,,D3DRM_OK Ž”,,,ff ,•, •’,^—,,,, Direct3D•Žf

```

lpDDS

```

DirectDraw fff
Ž IDirect3DRMTexture::InitFromFile,
IDirect3DRMTexture::InitFromResource
xe "IDirect3DRMTexture\:\SetColors"§
IDirect3DRMTexture::SetColors
HRESULT SetColors(DWORD ulColors);

```

```

fffff,f , ,ffff,•—
,,, fff ff fff,,,,
• E,,,D3DRM_OK Ž”,,,ff ,•, •’,^—
,,,, Direct3D•Žf f,•’ ,Ž ,,,,

```

ulColors

```

fffff’,8,,,
Ž IDirect3DRMTexture::GetColors
xe "IDirect3DRMTexture\:\SetDecalOrigin"§

```

IDirect3DRMTexture::SetDe•cale

```

xe "IDirect3DRMTexture\:\SetDecalSize"§

```

```

IDirect3DRMTexture::SetDecalSize
HRESULT SetDecalSize(D3DVALUE rvWidth, D3DVALUE
rvHeight);

```

```

• E,,,D3DRM_OK Ž”,,,ff ,•, •’,^—
,,,, Direct3D•Žf f,•’ ,Ž ,,,,

```

rvWidth, rvHeight

```

ff f, ,•, ,fff •,Ž’, fffff,fff,[1,1],,,

```

Ž IDirect3DRMTexture::GetDecalSize
xe "IDirect3DRMTexture\:\:SetDecalTransparency"\$
IDirect3DRMTexture::SetDecalTransparency
HRESULT SetDecalTransparency(BOOL bTransp);

□□□□○□□□□□□,□□,^—
,,,,,□□**Direct3D•Žf□f,•'□,Ž□,,,,□**

bTransp

ff□f,"-□,Ž,,,,,TRUE□□\□-□,Ž,,,,,FALSE,Ž',,□fffff',FALSE,,,□
Ž□□IDirect3DRMTexture::GetDecalTransparency
xe "IDirect3DRMTexture\:\:SetDecalTransparentColor"\$
IDirect3DRMTexture::SetDecalTransparentColor
HRESULT SetDecalTransparentColor(D3DCOLOR rcTransp);

ff f,"- , ' ,,
• D3DRM_OK Ž",,
Direct3D•Žf f,•' ,Ž ,,,,

rcTransp
,,,"- fffff,"- , ,,,
Ž□□IDirect3DRMTexture::GetDecalTransparentColor

**xe "IDirect3DRMTexture\
:\:SetShadesfffffŽ,fffff,Š□,Ž—,,,%o%o
,□',□,□',□**

fffff,,,fffff,%o%o,□',□,,,,,□,-",fff,,□,ffff,•-,,,,,□fff□f□fff,,,,,□
• D3DRM_OK ,ff ,: ,•,^—
,,,,, Direct3D•Žf f,•' ,Ž ,,,,

ulShades
%o%o, , ,',2,— ,,, ffff',16,,
Ž IDirect3DRMTexture::GetShades

IDirect3DRMUserVisual
ffff□fff,□Direct3DRMUserVisualfffff,□Š%o,,,,,IDirect3DRMUserVisual
ff,ffff,Ž—
,,□ffff□fff,IDirect3DRM::CreateUserVisualffff,☉,□,,,,□,□Š%o,•—
,,□□Š%o,•—,,□

GetClassName

GetName

SetAppData

SetName

Direct3DRMUserVisual

IDirect3DRM::CreateUserVisual

xe "IDirect3DRMUserVisu

al\:\Init"§

IDirect3DRMUserVisual::Init

HRESULT

Init(D3DRMUSERVISUALCAL

LBACK d3drmUVProc, void *

lpArg);

Direct3DRMUserVisualfffff, §

%o,,

•

Œ,,D3DRM_OK Ž”,,ff ,

•, •,^

—, **Direct3D の**

d3drmUVProc

D3DRMUSERVISUALCALLBACK

lpArg

∞∞

IDirect3DRM::CreateUserVisual

IDir

ect3DRMUserVisual::Init, Ⓒ, ,

•—

,,,, IDirect3DRM::CreateObj

ctffff,,,f f fffff

□f

IDirect3DRMViewport

IDirect3DRMViewport

∞∞

IDirect3DRMViewport

fff ffff,ffffff,,, Š”,,,,Š— GetBack

, IDirect3DRMViewport,IDirect

3DRMViewportArrayfff ffff ,

Ž ,,,

IDirect3DRMViewportfff ffff GetFront

ffff, ^%,ff f,*,,,,,,

fff

GetCamera

GetPlane

SetCamera

SetBack

SetFront

ffffff ff f

SetPlane

GetHeight

GetWidth

GetField

SetField

Init

の

Clear

Configure

ForceUpdate

GetDevice

tion

SetProjection

ff* □ *fff

GetUniformScaling

SetUniformScaling

•š

InverseTransform

Transform

IDirect3DRMViewportfff□ffff,□,,,,COMfff□ffff,“-
IUnknownfff□ffff□ffff,☉□,,□IUnknownfff□ffff ㉔

AddRef

QueryInterface

Release

IDirect3DRMViewport
IDirect3DRMObjectfff ffff,,ž,ffff,☉ ,,
AddDestroyCallback

Clone

DeleteDestroyCallback

GetAppData

GetClassName

GetName

SetAppData

SetName

Direct3DRMViewport *fffff*, \square IDirect3DRM::CreateViewport ①

```
xe "IDirect3DRMViewport\:\:Clear"§
IDirect3DRMViewport::Clear
HRESULT Clear();
```

①

• IDirect3DRMViewport\:\:Configure"§

IDirect3DRMViewport::Configure

```
HRESULT Configure(LONG lX, LONG lY, DWORD dwWidth,
DWORD dwHeight);
```

- D3DRM_OK \checkmark ff $,$ $,$ $,$ $^$ —
 $Direct3D$ \checkmark f $,$ $,$ \checkmark $,$

lX, lY

ff \square *f* \square ①

dwWidth, dwHeight

①

IX + dwWidth □□□□□の□□□□
D3DRMERR_BADVALUE,•,□

xe "IDirect3DRMViewport\:\:ForceUpdate"\$§

IDirect3DRMViewport::ForceUpdate

HRESULT ForceUpdate(DWORD dwX1, DWORD dwY1, DWORD dwX2,
DWORD dwY2);

ff□f□f, -^, <□", □□, , □Ž' , , , -^, □Ž%, IDirect3DRMDevice::Update の
”

- □E,,,D3DRM_OK Ž”,,,ff ,•, •’, ^—
,,,,, Direct3D•Žf f,•’ ,Ž ,,,,

dwX1, dwY1

—^, <, •

dwX2, dwY2

—^,%o%o<, •

ffff, □Ž’ ,,,<E,,,’, f□fff, □□,,,,,,□□□,,,,,□fff

xe "IDirect3DRMViewport\:\:GetBack"\$§

IDirect3DRMViewport::GetBack

D3DVALUE GetBack();

ff□f□

- □□□□□□□□□□ tCamera"\$§

IDirect3DRMViewport::GetCamera

HRESULT GetCamera(LPDIRECT3DRMFRAME *lpCamera);

ff□f□f, fff, Ž”, , □

- □E,,,D3DRM_OK □Ž” , , , ff□, •, □•’, ^—,,,,, □□Direct3D の

lpCamera

ff,•,Direct3DRMFrameffffff,“,• ,fff

Ž IDirect3DRMViewport::SetCamera, Camera

xe "IDirect3DRMViewport\:\:GetDevice"\$§

IDirect3DRMViewport::GetDevice

HRESULT GetDevice(LPDIRECT3DRMDEVICE *lpD3DRMDevice);

,,ffff,Ž“,,

• □□□□□ **D3DRM_OK**のDirect3D □□□□□
iewport\:\:GetDirect3DViewport"§

IDirect3DRMViewport::GetDirect3DViewport
HRESULT GetDirect3DViewport (LPDIRECT3DVIEWPORT *
 lpD3DViewport);

Direct3DRMViewport Direct3Dfffff,Ž“,,

- D3DRM_OK Ž”,,,ff ,*, *’^—
 ,,,, Direct3D•Žf f,•’ ,Ž ,,,,

lpD3DViewport

Direct3DViewportffffff,ffff,“,ffff

xe "IDirect3DRMViewport\:\:GetField"§
 IDirect3DRMViewport::GetField

D3DVALUE GetField();

Ž□□ **IDirect3DRMViewport::SetField,**
ViewinFrustum

xe "IDirect3DRMViewport\:\:GetFront"§

IDirect3DRMViewport::GetFront
 D3DVALUE GetFront();

ff□f□f,ffff□fffff□ff□f,^’,Ž”,,□

- ^,Ž’,□

Ž IDirect3DRMViewport::SetFront, Viewing Frustum

xe "IDirect3DRMViewport\:\:GetHeight"§

IDirect3DRMViewport::GetHeight

DWORD GetHeight();

ff f f, ,ffff’^,Ž“,,

- ,ffff’^’,*

xe "IDirect3DRMViewport\:\:GetPlane"§
IDirect3DRMViewport::GetPlane
HRESULT GetPlane(D3DVALUE *lpd3dvLeft, D3DVALUE *lpd3dvRight,

D3DVALUE *lpd3dvBottom, D3DVALUE *lpd3dvTop);

ffff□ffffff□ff□

- D3DRM_OKのDirect3D の

lpd3dvLeft, lpd3dvRight, lpd3dvBottom, lpd3dvTop

○○○

IDirect3DRMViewport::SetPlane

xe "IDirect3DRMViewport\:\:GetProjection"§

IDirect3DRMViewport::GetProjection

D3DRMPROJECTIONTYPE GetProjection();

f f,, • "%0,"Ž"%0,,,,,—
,,,,,,

- D3DRMPROJECTIONTYPE
—(E,fff,,,,,*,

Ž IDirect3DRMViewport::Se
tProjection

xe "IDirect3DRMViewport\
\:GetUniformScaling"§

IDirect3DRMViewport::GetUni
formScaling

BOOL GetUniformScaling();

fffff,fff,','•Žfff f,ff fff,,,
,—,,, Ž“ ,Ž“,

- ff f f,^,"—

```
,ff fff,,, ,TRUE ,,,,,,FA
LSE,*,
```

```
Ž IDirect3DRMViewport::Se
tUniformScaling
```

```
xe "IDirect3DRMViewport\
\GetWidth"§
```

```
IDirect3DRMViewport::GetWi
dth
```

```
DWORD GetWidth();
```

```
ff f f,*,ffff^,Ž“,,,
```

```
• *,ffff^*,
```

```
xe "IDirect3DRMViewport\
\GetX"§
```

```
IDirect3DRMViewport::GetX
```

```
LONG GetX();
```

```
ffff ,ff f fŽ“,x •ffff,Ž“
”
```

```
• xffff,*,
```

```
xe "IDirect3DRMViewport\
\GetY"§
```

```
IDirect3DRMViewport::GetY
```

```
LONG
```

G GetY());

ののの y

- y

xe "IDirect3DRMViewport\:\:Init"§

IDirect3DRMViewport::Init

HRESULT Init(LPDIRECT3DRMDEVICE lpD3DRMDevice,

LPDIRECT3

DRMFRAME

lpD3DRMFrameCamera,

DWORD xpos, DWORD ypos,

DWORD width, DWORD h

eight);

Direct3DRMViewport

- *D3DRM_OK*
Direct3D の

,

lpD3DRMDevice

*„ff f f,Š~•,,„DirectD3DRM
Devicefffff*

lpD3DRMFrameCamera

„ff f f,Š~•,,„fff ff f

xpos, ypos

ff f f, <x •,,y •

CE, ,, CE,,, IDirect3DRMPickedArrayfff ffff,—
CE,ffff,“,ffff,,ffff

xe "IDirect3DRMViewport\:\:Render"§
IDirect3DRMViewport::Render
HRESULT Render(LPDIRECT3DRMFRAME lpD3DRMFrame);

ff fŠ‘, Ž’,,,ff f f,ffffff,, Ž’,,,ff f ,fffff, ,%o
,Š‘,ff f,,,ffffff,,

• CE,,,D3DRM_OK Ž”,,,ff ,• ,•,^—,,, Direct3D
f f,• ,Ž’ ,,,,

lpD3DRMFrame
ffffff,,ff fŠ‘,Ž’,Direct3DRMFrameffffff,Ž’,• ,,ffff

xe "IDirect3DRMViewport\:\:SetBack"§
IDirect3DRMViewport::SetBack
HRESULT SetBack(D3DVALUE rvBack);

ff f f,fff fffffff ff f,^’, ’,,
• CE,,,D3DRM_OK Ž”,,,ff ,• ,•,^—
,,,,, Direct3D•Ž’f f,•’ ,Ž’ ,,,,

rvBack
fff fffffff ff f, ,^’
Ž’ IDirect3DRMViewport::GetBack, IDirect3DRMViewport::SetFront,
Viewing Frustum

xe "IDirect3DRMViewport\:\:SetCamera"§
IDirect3DRMViewport::SetCamera
HRESULT SetCamera(LPDIRECT3DRMFRAME lpCamera);

’,,
• □CE,,,D3DRM_OK□Ž”,,,ff□,•,□,•,^— Direct3D の

lpCamera
Direct3DRMFrame fff,Ž’,•
,,ffff, ff f f,^’ •CE CE,,Ž’,,,fff ff f,“,, ’,, ff , fff ff f, ,z
Ž’•CE,%o,, ,yŽ’, •CE,CE,,,

Ž’ IDirect3DRMViewport::GetCamera, Camera
xe "IDirect3DRMViewport\:\:SetField"§

IDirect3DRMViewport::SetField
 HRESULT SetField(D3DVALUE rvField);

ff f f,ff ,ff ff, ',,
 • □E,,D3DRM_OK□Ž”,,,ff□,•,□,^—,,,,,□□Direct3D•Žf□f,•' ,Ž

rvField

0.5 0 の

D3DRMERR_BADVALUE

□□ IDirect3DRMViewitFront"§

IDirect3DRMViewport::SetFront

HRESULT SetFront(D3DVALUE rvFront);

ff□f□f,ffff□fffff□ff□f,^',□',,,□

- D3DRM_OK
 Direct3D の

rvFront

の

1.0 0

の',“,,,, ,,ffff,D3DRMERR_BADVALUEff ,,
 Ž IDirect3DRMViewport::GetFront, Viewing Frustum
 xe "IDirect3DRMViewport\:\.SetPlane"§

IDirect3DRMViewport::SetPlane

HRESULT SetPlane(D3DVALUE rvLeft, D3DVALUE rvRight, D3DVALUE
 rvBottom,
 D3DVALUE rvTop);

fff,zŽ,',,ff□f□f,fff,fff□.□Ž”,,,ff□,•,□,^—
 ,,,,□□Direct3D•Žf□f,•'□,Ž□,,,,□

rvLeft, rvRight, rvBottom, rvTop

ff□f□f,□□, x□• □□', x□•□□□, y□•□□', y□•□

'%

, '□,,, ff□f□f,Ž', , IDirect3DRMViewport::SetFieldffff,^',□,ffff,,,,□”^,

“—, ^', ff□f□f,Ž', ,,,,,,□,,,□•—,Ž□',□', ,□%-□-

,ffff□ff□, ffffff,,,,,□

IIDirect3DRMViewport::GetPlane, IDirect3DRMViewport::SetField

xe "IDirect3DRMViewport::SetProjection"§

IDirect3DRMViewport::SetProjection

HRESULT SetProjection(D3DRMPROJECTIONTYPE rptType);

ff f f,Ž%fff, ',,

- CE,,,D3DRM_OK Ž",,,ff ,*, •',^—
,,,,, Direct3D•Žf f,•' ,Ž ,,,,

rptType

D3DRMPROJECTIONTYPE—(CE,fff,,,

Ž **IDirect3DRMViewport::GetProjection**

xe "IDirect3DRMViewport::SetUniformScaling"§

IDirect3DRMViewport::SetUniformScaling

HRESULT SetUniformScaling(BOOL bScale);

•Žfff f,ffffff,fff,,',,ff fff,,,,,—,,,, Ž“ , ',,

- CE,,,D3DRM_OK ^—
,,,,, Direct3D•Žf f,•' ,Ž ,,,,

bScale

,, Ž“ ,,,fff f,TRUE,Ž',,, ••CE, '•CE,“ , ŽCE ,Ž—

,, ,,,,,, •Žfff f,ffffff,^',,,,, ^; ∅
TRUE,,,

' ,ffff, fffffff,ff f,,,,,IDirect3DRMViewport::SetPlaneffff,,Ž—
'''

Ž IDirect3DRMViewport::GetUniformScaling

xe "IDirect3DRMViewport::Transform"§

IDirect3DRMViewport::Transform

HRESULT Transform(D3DRMVECTOR4D * lprvDst, D3DVECTOR *
lprvSrc);

lprvSrcffff f,Ž,ffff,f ff •,fff f •,Š, CE%,lprvDstfff f,•,

- CE,,,D3DRM_OK Ž",,,ff ,*, •',^—
,,,,, Direct3D•Žf f,•' ,Ž ,,,,

lprvDst

•Š‘□,□,,□^%o,•'Ž,□,,,,□CE%o,ffff,Ž,“,%oŽ□‘,,,□

μ §

IDirect3DRMWinDevice

```

ffff□fff,□ffffff□fff□ff,ffffff□fff□f,%",,,,,,IDirect3DRMW
inDevicefff□f のの
IDirect3DRMWinDevicefff ffff ffff,ffffff,, Š",,,,Š—
, ffffŠ— ,Ž ,,,,
IDirect3DRMWinDevicefff ffff ffff, ~%o,ff f,•—,,,,,,
HandleActivate

```

HandlePaint

```

IDirect3DRMWinDevicefff ffff, ,,,COMfff ffff,“—
IUnknownfff ffff ffff,CE ,, IUnknownfff ffff,Ž,,,,ffff,ff f
”
AddRef

```

QueryInterface

Release

```

Direct3DRMWinDevice □□□□□□□□
IID_IDirect3DRMWinDevice □□□□□
IDirect3DRM::QueryInterface □□□□□□□□□□
IDirect3DRM::CreMDevice,,CE □,,,□

```

**xe "IDirect3DRMWinDevice\
 \:HandleActivate"§**

```

IDirect3DRMWinDevice::HandleActivate
HRESULT HandleActivate(WORD wParam);

```

Windows の WM_ACTIVATE

»,•,,,,,,

— ffffff,ffff

Ž IDirect3DRM::CreateWrap

xe "IDirect3DRMWrap\:\:ApplyRelative"§

IDirect3DRMWrap::ApplyRelative

HRESULT ApplyRelative(LPDIRECT3DRMFRAME frame,
LPDIRECT3DRMOBJECT mesh);

fffff,“,fff, ’, ,Š“ff f,f ff•Š, , Ž,fff,Ž ff f, f ff•Š,
,,

• □□□□□ D3DRM_OKのDirect3D □□□□□

apfffff□

Ž□□ IDirect3DRM::CreateWrap

xe "IDirect3DRMWrap\:\:Init"§

IDirect3DRMWrap::Init

HRESULT Init(D3DRMWRAPTYPE d3drmw, LPDIRECT3DRMFRAME lpd3drmfRef,
D3DVALUE ox, D3DVALUE oy, D3DVALUE oz,

D3DVALUE dx, D3DVALUE dy, D3DVALUE dz,

D3DVALUE ux, D3DVALUE uy, D3DVALUE uz,

D3DVALUE ou, D3DVALUE ov, D3DVALUE su, D3DVALUE sv);

Direct3DRMWrapfffff, Š%,,

• ☉,,D3DRM_OK Ž”,,ff ,•, •’,^—
,,,, Direct3D•Žf f,•’ ,Ž ,,,

d3drmw

D3DRMWRAPTYPE の ff,,,

lpd3drmfRef

„Direct3DRMWrapfffff,Ž ff f,Ž,Direct3DRMFramefffff,ffff

ox, oy, oz

fff,☉“

dx, dy, dz

fff,zŽ

ux, uy, uz

y

ou, ov

su sv

```

    IDirect3DRMTOR min, max;
    }D3DRMBOX;
typedef D3DRMBOX *LPD3DRMBOX;

```

```

IDirect3DRMMesh::GetBox, IDirect3DRMMeshBuilder::GetBoxffff, Ž`,, <@`^, ' <,
, □
min, max
<@`^, ' <, ' □, , , , ' , D3DVECTOR □ ` ` , , , □
Ž □ □ D3DVECTOR, IDirect3DRMMesh::GetBox,
IDirect3DRMMeshBuilder::GetBox
xe "D3DRMIMAGE"§

```

D3DRMIMAGE

```

typedef struct _D3DRMIMAGE {
    int           width, height;
    int           aspectx, aspecty;
    int           depth;
    int           rgb;
    int           bytes_per_line;
    void*          buffer1;
    void*          buffer2;
    unsigned long   red_mask;
    unsigned long   green_mask;
    unsigned long   blue_mask;
    unsigned long   alpha_mask;
    int           palette_size;
    D3DRMPALETTEENTRY* palette;
}D3DRMIMAGE;
typedef D3DRMIMAGE, *LPD3DRMIMAGE;

```

IDirect3DRM::CreateTexture □□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□ **f f,•,□,□**

aspectx ,, **aspecty**

”□•**ffff,ffff**”□

depth

ffff“,,, *fff*□□
rgb
,, *fff*, FALSE, □□□*ffff*, *ffff*, *ffff**ffff*, Ž, □, ,, *fff*, TRUE, ,, ,, □*ffff*, *f*□*f*%
,, , RGB’, ,, , □
bytes_per_line
1*fff*“, ,, *fff*□□, ,, , 4, “□, ,, , □
buffer1
*ffff**fff*, □, ,, , *fff*□□□, *ffff*□, *ffff*□
buffer2
2□*ffff**ffff*, 2”-, *ffff**fff*□ NULL

red_mask, green_mask, blue_mask, alpha_mask

rgb, TRUE, ,, ,, *fff,ffff*, — •, *fff*, ,, , rgb, FALSE, ,, ,, *ffff*, —
, — , Š—‘*fff,fff*, ,, ,, , , SVGA*ffff*, Ž—, —
, “ ,64, ,, ,, ,, *fff,0xfc,fff*, ,,

palette_size

ffff□*ff*
palette
rgb, FALSE, ,, *fff,ffff* *ffff*, Ž, D3DRMPALETTEENTRY “, *fff*
f, ,,

Ž□□ **IDirect3DRM::CreateTexture**, IDirect3DRMTexture::GetImage

xe "D3DRMLOADMEMORY"§

D3DRMLOADMEMORY

typedef struct _D3DRMLOADMEMORY {

LPVOID lpMemory;

DWORD dSize;

} D3DRMLOADMEMORY, *LPD3DRMLOADMEMORY;

ffff **fff**, D3DRMLOAD_FROMMEMORY, Ž’, , IDirect3DRM::Load*fff*
f ,, , Load*ffff* ,•, ,, ,, , *f* ,, , *ff* *f*, Ž,

lpMemory

dSize

f *f*, ,, *fff* *ffff,fff,fff*^, Ž,

Ž Direct**3DRM::Load**, IDirect3DRMAnimationSet::Load,
IDirect3DRMFrame::Load, IDirect3DRMMeshBuilder::Load,
D3DRMLOADOPTIONS, D3DRMLOADRESOURCE

D3DRMIMAGE “,Ž—,,ff ffff,‘ , , , “, D3DRMIMAGE の
rgbfff,FALSE,Ž’,,,,,,,—,, TRUE,Ž’,,,,,,, RGB’,—,,

red, green, blue

ffff,’,Ⓔ —,’,’ 0,,255,”^,’Ž’,,,,,,,

flags

,,,,,Ž—,,,,,’,

D3DRMPALETTEFLAGS

のの

D3DRMIMAGE, D3DRMPALETTEFLAGS

xe "D3DRMPICKDESC"§

D3DRMPICKDESC

**typedef struct _D3DRMPICKDESC}
D3DRMPICKDESC, *LPD3DRMPICKDESC;**

IDirect3DRMPickedArray::GetPickffff,, , Ž“,, ,□fff,^’□-□ff□fŽ•Ž,Š“,,□

ulFaceIdx

Ž“,, ,fffff,-,fffff□

lGroupIdx

Ž“,, ,fffff,ff□fŽ•Ž□

vPosition

Ž“,, ,fffff,^’,Ž,’□,,’,D3DVECTOR□`\\,, ,□

Ž□□D3DVECTOR, IDirect3DRMPickedArray::GetPick

xe "D3DRMQUATERNION"§

D3DRMQUATERNION

typedef struct _D3DRMQUATERNION {

D3DVALUE s;

D3DVECTOR v;

}D3DRMQUATERNION;

typedef D3DRMQUATERNION, *LPD3DRMQUATERNION;

IDirect3DRMAnimation::AddRotateKeyffff,—

,,,%“,‘ , , , Direct3D, ŠŠ ,,,,,,Ž—,,

Ž IDirect3DRMAnimation::AddRotateKey,

D3DRMQuaternionFromRotation, D3DRMQuaternionMultiply,

D3DRMQuaternionSlerp, D3DRMMatrixFromQuaternion

xe "D3DRMVECTOR4D"§

D3DRMVECTOR4D

typedef struct _D3DRMVECTOR4D {

D3DVALUE x;

D3DVALUE y;

```

    D3DVALUE z;

    D3DVALUE w;
}D3DRMVECTOR4D;

typedef D3DRMVECTOR4D, *LPD3DRMVECTOR4D;

```

① IDirect3DRMViewport, Ž —, ,, □

x, y, z, w

D3DVALUE €, ' □ • Š € % 0 0, “, □

```

Ž □ □ IDirect3DRMViewport::Transform, IDirect3DRMViewport::InverseTransform
xe "D3DRMVERTEX"§
D3DRMVERTEX
typedef struct _D3DRMVERTEX{
    D3DVECTOR position;
    D3DVECTOR normal;

    D3DVALUE tu, tv;

    D3DCOLOR color;
} D3DRMVERTEX;

```

Direct3DRMMesh ①

position

normal

①

tu, tv

”“, •, ,, 'fffff •

color

”“,

Ž IDirect3DRMMesh::GetVertices, IDirect3DRMMesh::SetVertices

—<€

```

xe "D3DRMCOLORSOURCE"§

```

D3DRMCOLORSOURCE

```

typedef enum _D3DRMCOLORSOURCE{

```

D3DRMCOLOR_FROMFACE,

D3DRMCOLOR_FROMVERTEX

```
} D3DRMCOLORSOURCE;
```

```
Direct3DRMMeshBuilder::SetColorSource( D3DRMCOLORSOURCE_*, IDirect3DRMMeshBuilder::GetColorSource )
```

```
D3DRMCOLOR_FROMFACE
```

```
D3DRMCOLOR_FROMVERTEX
```

```
IDirect3DRMMeshBuilder::GetColorSource
```

```
xe "D3DRMCOMBINETYPE"§
```

```
D3DRMCOMBINETYPE
```

```
typedef enum _D3DRMCOMBINETYPE {  
    D3DRMCOMBINE_REPLACE,  
    D3DRMCOMBINE_BEFORE,  
    D3DRMCOMBINE_AFTER  
} D3DRMCOMBINETYPE;
```

```
D3DRMCOMBINE_REPLACE
```

```
0
```

```
D3DRMCOMBINE_BEFORE
```

```
1
```

```
D3DRMCOMBINE_AFTER
```

```
2
```

```
IDirect3DRMFrame::AddRotation, IDirect3DRMFrame::AddScale,  
IDirect3DRMFrame::AddTransform, IDirect3DRMFrame::AddTranslation
```

```
xe "D3DRMFILLMODE"§
```

```
D3DRMFILLMODE
```

```
typedef enum _D3DRMFILLMODE {
```

```
    D3DRMFILL_POINTS = 0 * D3DRMLIGHT_MAX,
```

```
    D3DRMFILL_WIREFRAME = 1 * D3DRMLIGHT_MAX,
```

```
    D3DRMFILL_MASK = 7 * D3DRMLIGHT_MAX,  
    D3DRMFILL_MAX = 8 * D3DRMLIGHT_MAX
```

```

} D3DRMFILLMODE;

D3DRMRENDERQUALITY, ' < , ž , , , - < , , , , □
D3DRMFILL_POINTS
" , , , " , , , , □ □ □ , " , , , , f □ f □
D3DRMFILL_WIREFRAME
f f f □ f f □ f , " , □
D3DRMFILL_SOLID
- \ f f f f f f , " , □
D3DRMFILL_MASK
f f f , ž - , , " , □
D3DRMFILL_MAX
" , , , , f □ f , □ \ ' □
ž □ □ D3DRMLIGHTMODE, D3DRMSHADEMODE, D3DRMRENDERQUALITY
xe "D3DRMFOGMODE"§
D3DRMFOGMODE
typedef enum _D3DRMFOGMODE{
    D3DRMFOG_LINEAR,
    D3DRMFOG_EXPONENTIAL,
    D3DRMFOG_EXPONENTIALSQUARED
} D3DRMFOGMODE;

f f f , , , < - , □ , , , , □ f f f % , , , , , \ , , , , , < , , , , ž , □
D3DRMFOG_LINEAR
f f f %

§
f f , , , , , , , ,
D3DRMFOG_EXPONENTIAL
f f f % , ž Š " , < , , , ž , < , ž , , , ,

§

```

D3DRMFOG_EXPONENTIALSQUARED

```

の 2 の
μ §
, , , , • ž , , , , < , , , , f f f ' , ' , , , f f f f f f , < , , , , ,
f f f , - " ž " " , IDirect3DRMFrame::SetSceneFogParams
e ∞∞

IDirect3DRMFrame::SetSceneFogMode,
IDirect3DRMFrame::SetSceneFogParams
xe "D3DRMFRAMECONSTRAINT"§
D3DRMFRAMECONSTRAINT
typedef enum _D3DRMFRAMECONSTRAINT {

```

D3DR_X

} D3DRMFRAMECONSTRAINT;

```
Direct3DRMFramefffff, •ž, , , , , <□", %", , ž, <□, , □, , -<Ⓔ, IDirect3DRMFrame::LookAtfffff, , , , , ž-, , , □
D3DRMCONSTRAIN_Z
x, , , y%", , ž-, , □
D3DRMCONSTRAIN_Y
x, , , z%", , ž-, , □
D3DRMCONSTRAIN_X
y, , , z%", , ž-, , □
ž□□IDirect3DRMFrame::LookAt
xe "D3DRMLIGHTMODE"§
D3DRMLIGHTMODE
typedef enum _D3DRMLIGHTMODE {
    D3DRMLIGHT_OFF    = 0 * D3DRMSHADE_MAX,
    D3DRMLIGHT_ON     = 1 * D3DRMSHADE_MAX,
    D3DRMLIGHT_MASK   = 7 * D3DRMSHADE_MAX,
    D3DRMLIGHT_MAX    = 8 * D3DRMSHADE_MAX
} D3DRMLIGHTMODE;

D3DRMRENDERQUALITYⒺ, 'Ĺ, , , -< ①
D3DRMLIGHT_OFF

    D3DRMLIGHT_ON
    ⒺⒺ, ff, , ,
D3DRMLIGHT_MASK

    D3DRMLIGHT_MAX
    ⒺⒺf f, '

ž□□D3DRMFILLMODE, D3DRMSHADEMODE,
D3DRMRENDERQUALITY

TYPE{

    D3DRMLIGHT_AMBIENT,

    D3DRMLIGHT_POINT,
    D3DRMLIGHT_SPOT,
    D3DRMLIGHT_DIRECTIONAL,
    D3DRMLIGHT_PARALLELPOINT
} D3DRMLIGHTTYPE;

IDirect3DRM::CreateLightfffff, Ⓔ, □, , □ⒺⒺfff, ' < , , □
D3DRMLIGHT_AMBIENT
```

```

fffff@E
D3DRMLIGHT_POINT
ffff@E
D3DRMLIGHT_SPOT
fffff@E
D3DRMLIGHT_DIRECTIONAL
fffff@E
D3DRMLIGHT_PARALLELPPOINT
fffff@E

xe "D3DRMMATERIALMODE"$
D3DRMMATERIALMODE
typedef enum _D3DRMMATERIALMODE{
    D3DRMMATERIAL_FROMMESH,
    D3DRMMATERIAL_FROMPARENT,
    D3DRMMATERIAL_FROMFRAME
} D3DRMMATERIALMODE;

```

IDirect3DRMFrame::GetMaterialMode
IDirect3DRMFrame::SetMaterialMode
D3DRMMATERIAL_FROMMESH

① ①

```

“,,, ,,,ffff ’,,
D3DRMMATERIAL_FROMPARENT
ffff@•,,,@ff@f,,@E@,,,
D3DRMMATERIAL_FROMFRAME

```

IDirect3DRMFrame::GetIMode
xe "D3DRMPALETTEFLAGS"\$

```

D3DRMPALETTEFLAGS
typedef enum _D3DRMPALETTEFLAGS {
    D3DRMPALETTE_FREE,
    D3DRMPALETTE_READONLY,
    D3DRMPALETTE_RESERVED
} D3DRMPALETTEFLAGS;

D3DRMPALETTEENTRY``,`@,ž-•-,'<,,@
D3DRMPALETTE_FREE
ffff@,,,ffff,ž-,ž-,,,@
D3DRMPALETTE_READONLY
@',,,,,,@ffff@,ž,,,,,,@
D3DRMPALETTE_RESERVED
ffff@,ž,,,,,,@
ž@D3DRMPALETTEENTRY
xe "D3DRMPROJECTIONTYPE"$

```

D3DRMPROJECTIONTYPE

```
typedef enum _D3DRMPROJECTIONTYPE {
    D3DRMPROJECT_PERSPECTIVE,
    D3DRMPROJECT_ORTHOGRAPHIC
} D3DRMPROJECTIONTYPE;
```

```
Direct3DRMViewport                               Ž%0fff,'<, , ,—<€
IDirect3DRMViewport::GetProjection, IDirect3DRMViewport::SetProjection
```

```
    D3DRMPROJECT_PERSPECTIVE
    “Ž“%0
D3DRMPROJECT_ORTHOGRAPHIC
```

□ □ IDirect3DRMViewport::GetProjection, IDirect3DRMViewpRMRENDERQUALITY

```
typedef enum _D3DRMSHADEMODE {
    D3DRMSHADE_FLAT           = 0,
    D3DRMSHADE_GOURAUD        = 1,
    D3DRMSHADE_PHONG          = 2,
    D3DRMSHADE_MASK           = 7,
    D3DRMSHADE_MAX            = 8
} D3DRMSHADEMODE;

typedef enum _D3DRMLIGHTMODE {
    D3DRMLIGHT_OFF            = 0 * D3DRMSHADE_MAX,
    D3DRMLIGHT_ON             = 1 * D3DRMSHADE_MAX,
    D3DRMLIGHT_MASK           = 7 * D3DRMSHADE_MAX,
    D3DRMLIGHT_MAX            = 8 * D3DRMSHADE_MAX
} D3DRMLIGHTMODE;

typedef enum _D3DRMFILLMODE {
    D3DRMFILL_POINTS          = 0 * D3DRMLIGHT_MAX,
    D3DRMFILL_WIREFRAME       = 1 * D3DRMLIGHT_MAX,
    D3DRMFILL_SOLID           = 2 * D3DRMLIGHT_MAX,
    D3DRMFILL_MASK            = 7 * D3DRMLIGHT_MAX,
    D3DRMFILL_MAX             = 8 * D3DRMLIGHT_MAX
} D3DRMFILLMODE;

typedef DWORD D3DRMRENDERQUALITY;

#define D3DRMRENDER_WIREFRAME
(D3DRMSHADE_FLAT+D3DRMLIGHT_OFF+D3DRMFILL_WIREFR
AME)
```



```

typedef enum _D3DRMUSERVISUALREASON {
    D3DRMUSERVISUAL_CANSEE,
    D3DRMUSERVISUAL_RENDER
} D3DRMUSERVISUALREASON;

ffff, D3DRMUSERVISUALCALLBACK fffffŠ, Ğ, □, —, ' < , , □
D3DRMUSERVISUAL_CANSEE
f□f□ffffffffff, ff□f□f□, %ž□\ , , , , □f□ffffŠ□, TRUE, • , , , , , , , □
D3DRMUSERVISUAL_RENDER
f□ffffŠ□, f□f□fffff□fffff, ffffff, , , , , , , , □
ž□□D3DRMUSERVISUALCALLBACK
xe "D3DRMWRAPTYPE"$
D3DRMWRAPTYPE
typedef enum _D3DRMWRAPTYPE{
    D3DRMWRAP_FLAT,
    D3DRMWRAP_CYLINDER,
    D3DRMWRAP_SPHERE,
    D3DRMWRAP_CHROME
} D3DRMWRAPTYPE;

IDirect3DRM::CreateWrap fffff, , , , □□, , , Direct3DRMWrap ffffff, fff, ' < , , □, , I
Direct3DRMWrap::Init fffff, Ğ, □, , ' , □Direct3DRMWrap ffffff, □š%, , , , , , , - < Ğ, ž
- , , , , , , □
D3DRMWRAP_FLAT
• - fff□
D3DRMWRAP_CYLINDER
%' fff□
D3DRMWRAP_SPHERE
< \ fff□
D3DRMWRAP_CHROME
fff, □fffff, ffffff□, "ž, , , , , fffff□•, š, ", , , □
ž□□IDirect3DRM::CreateWrap, IDirect3DRMWrap::Init,
IDirect3DRMWrap fff□ffff
xe "D3DRMXOFFORMAT"$
D3DRMXOFFORMAT
typedef enum _D3DRMXOFFORMAT{
    D3DRMXOF_BINARY,
    D3DRMXOF_COMPRESSED,
    D3DRMXOF_TEXT
} D3DRMXOFFORMAT;

IDirect3DRMMeshBuilder::Save fffff, ž- , , , fffff□ffff, ' < , , □
D3DRMXOF_BINARY
ffff, fffff□ff□fff, , , □, , , fffff□' , , , □
D3DRMXOF_COMPRESSED
Ğ□, ff□f, , , , , □
D3DRMXOF_TEXT
ffff, fffff□ff□fff, , , □
D3DRMXOF_BINARY, D3DRMXOF_TEXT□' , "ž, ž- , , , , , , , □
ž□□IDirect3DRMMeshBuilder::Save
xe "D3DRMZBUFFERMODE"$
D3DRMZBUFFERMODE
typedef enum _D3DRMZBUFFERMODE {
    D3DRMZBUFFER_FROMPARENT,

```

```

        D3DRMZBUFFER_ENABLE,
        D3DRMZBUFFER_DISABLE
    } D3DRMZBUFFERMODE;

Zffff,-E,,,,Ž,□
D3DRMZBUFFER_FROMPARENT
ff□f,□ff□f,,Zffff,□',E□,,□,,,ffff□',,,,□
D3DRMZBUFFER_ENABLE
Zffff,-E,,,,□
D3DRMZBUFFER_DISABLE
Zffff,-E,,,,□
Ž□□IDirect3DRMFrame::GetZbufferMode, IDirect3DRMFrame::SetZbufferMode
,,',E

```

```

xe "D3DRMANIMATIONOPTIONS"§
D3DRMANIMATIONOPTIONS
typedef DWORD D3DRMANIMATIONOPTIONS;
#define D3DRMANIMATION_CLOSED          0x02L
#define D3DRMANIMATION_LINEARPOSITION  0x04L
#define D3DRMANIMATION_OPEN            0x01L
#define D3DRMANIMATION_POSITION        0x00000020L
#define D3DRMANIMATION_SCALEANDROTATION 0x00000010L

#define D3DRMANIMATION_SPLINEPOSITION  0x08L

```

IDirect3DRMAnimation::GetOptions

IDirect3DRMAnimation::SetOptions

```

        D3DRMANIMATION_CLOSED
        fff□fff,E,•,□□,,□□“,’,,ŠŽ“,-,,f□f,,□•,,のののの
        ,,E,•,f□,—,—,□f□f,,f ののの

```

D3DRMANIMATION_LINEARPOSITION

```

        fff fff,~, E, ',,,

```

D3DRMANIMATION_OPEN

```

        1          'Ž,,

```

D3DRMANIMATION_POSITION

```

        のの

```

””

D3DRMANIMATION_SCALEANDROTATION

```

        のの

```

```

        ',,,,,•Š —, ,,,,

```

D3DRMANIMATION_SPLINEPOSITION

```

        fff fff,~,fffff, ',,,

```

xe "D3DRMCOLOMODEL"§

D3DRMCOLOMODEL

```

typedef D3DCOLOMODEL D3DRMCOLOMODEL;

```

D3DCOLORMODEL**□□ D3DCOLOPTIONS**

```

typedef DWORD D3DRMLOADOPTIONS;
#define D3DRMLOAD_FROMFILE           0x00L
#define D3DRMLOAD_FROMRESOURCE      0x01L
#define D3DRMLOAD_FROMMEMORY        0x02L
#define D3DRMLOAD_FROMSTREAM        0x03L
#define D3DRMLOAD_BYNAME             0x10L
#define D3DRMLOAD_BYPOSITION        0x20L
#define D3DRMLOAD_BYGUID             0x30L
#define D3DRMLOAD_FIRST              0x40L
#define D3DRMLOAD_INSTANCEBYREFERENCE 0x100L

#define D3DRMLOAD_INSTANCEBYCOPYING 0x200L

```

IDirect3DRM::Load□**IDirect3DRMAnimationSet::Load□****IDirect3DRMFrame::Load□□□□****IDirect3DRMMeshBuilder::Load □□□□□□,□*****f□f□fff*****D3DRMLOAD_FROMFILE**

```

ffff,,f□f,,□,,,fffff□',,,,□
D3DRMLOAD_FROMRESOURCE
f□f□f,,f□f,,□,,fff,Ž',,□□□Loadffff,⊕,□,, lpvObjSourcefff□f,□D3DRMLOADRE
SOURCE□'\,Ž,,,,,,,,□
D3DRMLOAD_FROMMEMORY
fff,,f□f,,□,,,fff,Ž',,□□□Loadffff,⊕,□,, lpvObjSource
fff□f,□D3DRMLOADMEMORY□'\,Ž,,,,,,,,□
D3DRMLOAD_FROMSTREAM
fff□f,,f□f,,□
Ž•Žfff
D3DRMLOAD_BYNAME

```

D3DRMLOAD_BYPOSITION

```

0の
0n -
,ffffff,f f,, ffff fff,ffffff,'ffffff,Š,,,,,,,,, 'ffffff,Š,,,,,,,,

```

D3DRMLOAD_BYGUID

```
GUID ,Ž—,ffffff,f f,,
```



```

xe "D3DRMMATRIX4D"§
D3DRMMATRIX4D
    typedef D3DVALUE D3DRMMATRIX4D[4][4];

    Š, —,,•, —ffff, D3DRMMATRIX4D[row][column],, ,,

typedef DWORD D3DRMSAVEOPTIONS;
#define D3DRMXOFSAVE_NORMALS 1
#define D3DRMXOFSAVE_TEXTURECOORDINATES 2
#define D3DRMXOFSAVE_MATERIALS 4
#define D3DRMXOFSAVE_TEXTURENAMES 8
#define D3DRMXOFSAVE_ALL 15
#define D3DRMXOFSAVE_TEMPLATES 16

IDirect3DRMMeshBuilder::Saveffff,fffff,'<,,□
D3DRMXOFSAVE_NORMALS

D3DRMXOFSAVE_TEXTURECOORDINATES

D3DRMXOFSAVE_MATERIALS

    D3DRMXOFSAVE_TEXTURENAMES
    Š–“,-,%o, fffff,-‘,•‘,,
    D3DRMXOFSAVE_ALL
    Š–“,-,%o, – ffff fffff • fffff fffff,-‘,•‘,,
D3DRMXOFSAVE_TEMPLATES

Direct3D OCOM ○
D3DRM_OK

D3DRMERR_BADALLOC

```

D3DRMERR_BADDEVICE

ffff,ffff ,ĀŠ ,,,

D3DRMERR_BADFILE

f ffff,•

D3DRMERR_BADMAJORVERSION

DLL,fff f fff,•

D3DRMERR_BADMINORVERSION

DLL,fff f fff,•

D3DRMERR_BADOBJECT

^ ,,ffffff,•—

D3DRMERR_BADTYPE

• ,Ā,^ ,“,,

D3DRMERR_BADVALUE

• ,’;^ ,“,,

D3DRMERR_FACEUSED

-,ffff ,Ž— ,,,,

D3DRMERR_FILENOTFOUND

ffff,Š,,,,,,

D3DRMERR_NOTDONEYET

Ž',,,,,,

D3DRMERR_NOTFOUND

Ž',,,•-,ffffff,€,,,,,

D3DRMERR_UNABLETOEXECUTE

fff ff,Ž□,,,,□