

ゲームプログラミング

DirectX Audio - 第13回 3Dサウンドバッファの設定

DirectSound3DBufferは、サウンドバッファの再生状態にかかわらず、位置、速度、コーンなどの変更をリアルタイムに行うことができます。これらの設定は、バッファごとに行えます。

位置の設定

サウンドバッファの位置の設定は、DirectSound3DBufferオブジェクトのSetPositionメソッドで行います。また、GetPositionメソッドで現在の位置を取得することができます。

```
// 位置設定(IpDS3DBufferは初期化済みのDirectSound3DBufferオブジェクト)  
IpDS3DBuffer8->SetPosition(10.0f, 0.0f, -4.0f, DS3D_IMMEDIATE); // 左からx, y, z位置
```

速度の設定

サウンドバッファの速度の設定は、DirectSound3DBufferオブジェクトのSetVelocityメソッドで行います。また、GetVelocityメソッドで現在の速度を取得することができます。

```
// 速度設定  
IpDS3DBuffer8->SetVelocity(1.0f, 0.0f, 0.0f, DS3D_IMMEDIATE); // 左からx, y, zの速度ベクトル
```

この速度は、DirectSound3Dがドップラー効果を計算するための値であり、実際にバッファが移動するわけではありません。

最小距離と最大距離の設定

DirectSound3Dでは、リスナーがサウンドバッファに近づくにつれて音量が大きくなり、距離が半分になると音量は倍になります。しかし、ある距離より近づいたときに音量が増加し続けるのは不都合となる場合があります。最小距離とは、この「ある距離」のことです。最小距離より近づいても音量は大きくなり、最小距離より離れると音量が小さくなります。

最小距離は、DirectSound3DBufferオブジェクトのSetMinDistanceメソッドで行います。デフォルト値はDS3D_DEFAULTMINDISTANCEで、1メートルと定義されています。

```
// 最小距離を0.02メートル(2cm)に設定  
IpDS3DBuffer8->SetMinDistance(0.02f, DS3D_IMMEDIATE);
```

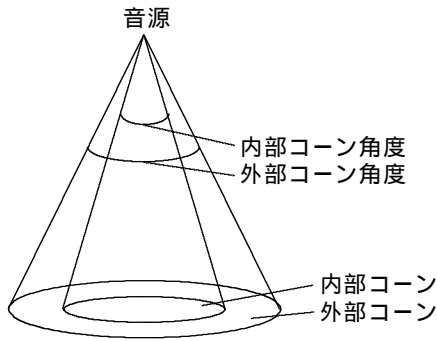
最小距離のほかに最大距離も設定できます。最大距離とは、そこを超えるとそれ以上サウンドの音量が小さくならない距離のことです。最大距離は、DirectSound3DBufferオブジェクトのSetMaxDistanceメソッドで行います。デフォルト値はDS3D_DEFAULTMAXDISTANCEで、10億メートルと定義されています。

```
// 最大距離を100メートルに設定  
IpDS3DBuffer8->SetMinDistance(100.0f, DS3D_IMMEDIATE);
```

DirectSound3Dでは、リスナーとサウンドバッファの距離が最小距離から最大距離の間にあるとき、音量が変化します。

サウンドコーンの設定

サウンドコーンとは、方向のあるサウンドを表すためのモデルのことです。サウンドコーンは、Direct3Dのスポットライトのような形になっており、内部コーンと外部コーンで形成されます。リスナーが内部コーンの中にいる場合、サウンドの音量はリスナーとの距離や向きを考慮したものになります。内部コーンの外側かつ外部コーンの内側である場合は、設定された係数によってサウンドの音量がさらに小さくなります。コーンから完全に外れた場合は、サウンドは聞こえなくなります。



コーンの向きはDirectSound3DBuffer8オブジェクトのSetConeOrientationメソッド、コーンの角度はSetConeAnglesメソッド、外部コーンの音量(の減衰量)はSetConeOutsideVolumeメソッドで行います。また、コーンのそれぞれの設定値はGetConeOrientationメソッド、GetConeAngleメソッド、GetConeOutsideVolumeメソッドで取得することができます。

```
// コーン設定
lpDS3DBuffer8->SetConeOrientation(0.0f, 0.0f, 1.0f, DS3D_IMMEDIATE); // 引数は方向ベクトル
lpDS3DBuffer8->SetConeAngles(90, 120, DS3D_IMMEDIATE); // 左から内部、外部コーンの角度
lpDS3DBuffer8->SetConeOutsideVolume(-1000, DS3D_IMMEDIATE); // 外部コーンでは10dB減衰
```

すべてのパラメータの設定

DirectSound3DBufferオブジェクトのSetAllParametersメソッドは、3Dサウンドバッファに関するすべてのパラメータの設定を行うことができ、GetAllParametersメソッドですべてのパラメータを取得することができます。なお、これらのメソッドを呼び出す前に、引数で渡すDS3DBUFFER構造体のdwSizeメンバに構造体のサイズを設定しておく必要があります。

```
// すべてのパラメータの設定
DS3DBUFFER ds3dBuf;
ZeroMemory(&ds3dBuf, sizeof(ds3dBuf));
ds3dBuf.dwSize = sizeof(ds3dBuf);
ds3dBuf.vPosition = D3DXVECTOR3(1.0f, 1.0f, 0.0f); // 位置
ds3dBuf.vVelocity = D3DXVECTOR3(0.0f, 0.0f, 1.0f); // 速度ベクトル
ds3dBuf.dwInsideConeAngle = 90; // 内部コーン角度
ds3dBuf.dwOutsideConeAngle = 120; // 外部コーン角度
ds3dBuf.vConeOrientation = D3DXVECTOR3(0.0f, 0.0f, -1.0f); // 方向ベクトル
ds3dBuf.lConeOutsideVolume = -100; // 外部コーンでの減衰量
ds3dBuf.flMinDistance = 0.5f; // 最小距離
ds3dBuf.flMaxDistance = 1000.0f; // 最大距離
ds3dBuf.dwMode = DS3DMODE_NORMAL; // 操作モード
lpDS3DBuffer8->SetAllParameters(&ds3dBuf, DS3D_IMMEDIATE);
```

課題

3Dバッファの設定を行う関数を完成させ、追加しましょう。

(1)以下のプログラムは、3Dバッファの位置を設定するDXASet3DPosition関数です。引数は左からバッファの指定、バッファの位置です。

```
/*
*****
*/
3 Dバッファ位置設定
*****
void DXASet3DPosition(const DXA_SB dxaBuf, const D3DVECTOR& dvPos)
{
    if(NULL == g_lpDS3DBuf8[dxaBuf]) {
        OutputDebugString("**** Error - 3 Dバッファ未生成(DXASetPosition)¥n");
        return;
    }
    g_lpDS3DBuf8[dxaBuf]->SetPosition(??????, ??????, ??????, DS3D_IMMEDIATE);
}
```

(2)以下のプログラムは、3Dバッファの速度を設定するDXASet3DVelocity関数です。引数は左からバッファの指定、バッファの速度ベクトルです。

```
/*
 3 D バッファ速度設定
*/
void DXASet3DVelocity(const DXA_SB dxaBuf, const D3DVECTOR& dvVelocity)
{
    if(NULL == g_lpDS3DBuf8[dxaBuf]) {
        OutputDebugString("*** Error - 3 D バッファ未生成(DXASetVelocity)¥n");
        return;
    }
    g_lpDS3DBuf8[dxaBuf]-> ここは各自考えましょう
}
```

(3)以下のプログラムは、3Dバッファの最小距離と最大距離を設定するDXASet3DDistance関数です。引数は左からバッファの指定、最小距離、最大距離です。

```
/*
 3 D バッファ減衰距離設定
*/
void DXASet3DDistance(const DXA_SB dxaBuf, const D3DVALUE dvMin, const D3DVALUE dvMax)
{
    if(NULL == g_lpDS3DBuf8[dxaBuf]) {
        OutputDebugString("*** Error - 3 D バッファ未生成(DXASetDistance)¥n");
        return;
    }
    g_lpDS3DBuf8[dxaBuf]-> ここは各自考えましょう
    g_lpDS3DBuf8[dxaBuf]-> ここは各自考えましょう
}
```

(4)以下のプログラムは、3Dバッファのコーン(コーンの方向、内部コーン角度、外部コーン角度、外部コーンの音量)を設定するDXASet3DConeParameters関数です。引数は左からバッファの指定、コーンの方向ベクトル、内部コーン角度、外部コーン角度、外部コーンの音量(の減衰量)です。

```
/*
 3 D バッファコーン設定
*/
void DXASet3DConeParameters(const DXA_SB dxaBuf, const D3DVECTOR& dvOrientation,
                             const DWORD dwInsideAngle, const DWORD dwOutsideAngle,
                             const long lOutsideVolume)
{
    if(NULL == g_lpDS3DBuf8[dxaBuf]) {
        OutputDebugString("*** Error - 3 D バッファ未生成(DXASetConeParameters)¥n");
        return;
    }
    g_lpDS3DBuf8[dxaBuf]-> ここは各自考えましょう
    g_lpDS3DBuf8[dxaBuf]-> ここは各自考えましょう
    g_lpDS3DBuf8[dxaBuf]-> ここは各自考えましょう
}
```

(5)以下のプログラムは、3Dバッファのすべてのパラメータを設定するDXASet3DAllParameters関数です。引数は左からバッファの指定、3Dバッファの情報が格納されたDS3DBUFFER構造体の変数です。

```
/*
 3 D バッファ全パラメータ設定
*/
void DXASet3DAllParameters(const DXA_SB dxaBuf, const DS3DBUFFER& ds3DBuf)
{
    if(NULL == g_lpDS3DBuf8[dxaBuf]) {
        OutputDebugString("*** Error - 3 D バッファ未生成(DXASetAllParameters)¥n");
        return;
    }
    g_lpDS3DBuf8[dxaBuf]-> ここは各自考えましょう
}
```