Use our 3D visualizer with your own software with MEVP.dll and our Intelligent Dmx Interface

Overview

Our MEVP.dll is a 32 bit Windows DLL (Dynamic Link Librairy). and works on Windows ME, 2000 and XP. It has been tested on Visual C++.

Files

The required files are :

- MEVP.h
- MEVP.dll

Functions prototypes

The MEVP.dll contains four functions :

- **int DasMevStart(char* sController, char* sPassWrd)**

  Start the 3D visualizer and open shared memory.

  Parameters:

  sController Your software's name

  sPassWrd Your password

  Return value:

  1 if succeeds, **0** if failed

- **int DasMevCommand(int iType, int iParam)**

The first parameter <***iType***> defines the thing to do :

iType	*explanation*	*iParam*	*return value*
MEVP_CLOSE_VISUALIZER	Close the 3D visualizer	not used	1 if the function succeeds,0 if the function failed
MEVP_SET_LANGUAGE	Set the language	Specifies the language to set	1 if the function succeeds,0 if the function failed
MEVP_READ_PATCH	Specifies to the 3D visualizer that It has to write its fixture's patch	not used	The number of fixtures written if the function succeeds,0 if the function failed

- **int DasMevWriteDmx(int iUniverse, BYTE* DmxArray)**

  Write a DMX universe to the shared memory.

  Parameters:

iUniverse	Index of the universe to write (must be < MEVP_NB_UNIVERSES)
DmxArray	DMX data array (length = MEVP_DMX_MAX_CHANNEL)

  Return value:

  1 if succeeds, **0** if failed

- **int DasMevGetFixtureParam(int iIndex, int* iDmxAddress, int* iDmxUniverse,**
  int* iNbChannels, char *sName,
  float* fPosX, float* fPosY, float* fPosZ,
  float* fRotX, float* fRotY, float* fRotZ)

  Get the fixture's patch parameters.
  Must be called after DasMevCommand(MEVP_READ_PATCH, 0) to inform the 3D visualizer that It has to write its fixture's patch.

  Parameters:

iIndex	Index of the fixture to read
iDmxAddress	DMX address of the fixture
DmxUniverse	DMX universe of the fixture
iNbChannels	Number of DMX channels used by the fixture
sName	Name of the fixture
fPosX	X axis coordinate of the fixture
fPosY	Y axis coordinate of the fixture
fPosZ	Z axis coordinate of the fixture
fRotX	X axis rotation of the fixture
fRotY	Y axis rotation of the fixture
RotZ	Z axis rotation of the fixture

  Return value:

  1 if succeeds, **0** if failed

- **int DasMevGetVersion(void)**

  Returns the DLL version.

Exemple of code using our DLL - C++ style

Write 1st DMX universe on shared memory:

```
BYTE ucDmxArray[512];

/* Fill the DMX buffer */

DasMevWriteDmx(0, ucDmxArray);
```

Get fixture's parameters of the 3D visualizer:

```
int iDmxAddress, iDmxUniverse, iNbChannels;
float fPosX, fPosY, fPosZ, fRotX, fRotY, fRotZ;
char sName[MEVP_MAX_STR_LEN];

int iNbFixt = DasMevCommand(MEVP_READ_PATCH, 0);

if (iNbFixt > 0)
{
   for (int i=0; i<iNbFixt; i++)
   {
        if (DasMevGetFixtureParam(i, &iDmxAddress, &iDmxUniverse,
                                     &iNbChannels, sName,
                                     &fPosX, &fPosY, &fPosZ,
                                     &fRotX, &fRotY, &fRotZ))
        {
             /* You can use fixture's parameters */
        }
   }
}
```

Please report any problems to marc@nicolaudie.com