

IKALiZER Manager & ioUrbanTerror

Source code upgrade

Copier les fichiers dans / Copy these files into : /code/client/

sam.c
sam.h
sam_lib.h

Modifier le fichier dans / Edit this file in : /code/client/

snd_main.c

Insérer le code suivant dans la fonction *S_Init* :

Insert this code in *S_Init* function :

```
// iKALiZER Manager
{
    long lReturn;
    long lInit;
    lReturn = Cvar_Flags ( "s_useSAM" );
    if (lReturn!=CVAR_NONEXISTENT)
    {
        char szTmp[16];
        cv = Cvar_Get ( "s_useSAM", "0", CVAR_ARCHIVE );
        lInit = cv->integer;
        cv->flags = CVAR_TEMP;

        sprintf ( szTmp, "%d", lInit );
        cv = Cvar_Get ( "s_useIkalizer", szTmp, CVAR_ARCHIVE );
    }
    else
    {
        cv = Cvar_Get ( "s_useIkalizer", "0", CVAR_ARCHIVE );
    }

    if (cv->integer && (!started) )
    {
        started = S_SAM_Init( &si );
        Cvar_Set( "s_backend", "IKALIZER" );
    }
}
```

Emplacement / Place :

Après / After : `Cmd_AddCommand("s_info", S_SoundInfo);`

<...> *Placer le code ici / Place core here*

Avant / Before : `cv = Cvar_Get("s_useOpenAL", "0", CVAR_ARCHIVE);`

Ajouter les fichiers au projet quake3 dans / Add these files to quake3 project in : /code/client/

sam.c
sam.h
sam_lib.h

Modifier cette ligne dans le fichier *files.c* / Update this line in *files.c* file : /code/qcommon

Avant / Before :

```
static int fs_numServerPaks;
```

Après / After :

```
int fs_numServerPaks;
```

Gestion du mode idle avec ioUrbanTerror / ioQuake

Idle mode management with ioUrbanTerror / ioquake

Modifier le fichier dans / Edit this file in : /code/qcommon/
common.c

Dans la fonction *Com_Frame* / In *Com_Frame* function

1 – Remplacer ce code / Replace this code

```
do {  
    ...  
} while ( msec < minMsec );  
Cbuf_Execute ();
```

2 – Avec ceci / With this

```
#ifdef _WIN32  
{  
    long iCount;  
    iCount = 0;  
    do {  
        com_frameTime = Com_EventLoop();  
        if ( lastTime > com_frameTime ) {  
            lastTime = com_frameTime;           // possible on first frame  
        }  
        msec = com_frameTime - lastTime;  
  
        // add idle time  
        iCount = (iCount+1)&15;  
        if (iCount==15) Sleep ( 1 );  
        else if (iCount>2) Sleep ( 0 );  
    } while ( msec < minMsec );  
}  
#else  
do {  
    com_frameTime = Com_EventLoop();  
    if ( lastTime > com_frameTime ) {  
        lastTime = com_frameTime;           // possible on first frame  
    }  
    msec = com_frameTime - lastTime;  
} while ( msec < minMsec );  
#endif  
Cbuf_Execute ();
```