

## Costruzione di Interfacce Lezione 20 MFC Opengl e Texture

[cignoni@iei.pi.cnr.it](mailto:cignoni@iei.pi.cnr.it)  
<http://vcg.iei.pi.cnr.it/~cignoni>

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

1

## Moebius4

- ❖ Partiamo nuovamente da zero
  - ❖ App Mfc, MDI
  - ❖ Cambiamo I nomi alle classi...
  - ❖ Deriviamo la classe vista da CFormView
  - ❖ Appena fatto il progetto aggiungiamo al progetto una classe derivata genericamente da Cview e chiamiamola CMBGLView
  - ❖ Sistemiamo include e tipi in modo che tutti conoscano i tipi che servono...

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

2

## CFormView

- ❖ Una form view è una view che contiene control (bottoni, combobox, listbox, tree ecc)
- ❖ I controlli sono piazzati secondo una risorsa dialog-template
- ❖ Ad ogni controllo nel template si può associare un oggetto, membro della classe che viene creato/piazzato/distrutto automaticamente, ma i cui eventi devono essere gestiti dall'utente

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

3

## Doppia View

- ❖ Obiettivo un'applicazione con da una parte opengl e dall'altra un dialogo.
- ❖ Si usa la classe CSplitterWnd
- ❖ Si modifica la classe CChildFrame (il frame della finestra padre della vista in un app mdi)
- ❖ In un sdi si modificava semplicemente Cmainframe in modo del tutto analogo a quanto vedremo.

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

4

## CMDIChildWnd

- ❖ Offre funzionalità di una child window in una multiple document interface (MDI).
- ❖ una MDI child window assomiglia a una tipica frame window, eccetto che la MDI child window appare dentro una MDI frame window piuttosto che sul desktop.
- ❖ Una MDI child window non ha una propria barra menu, but condivide i menu della MDI frame window.
- ❖ Il framework cambia automaticamente il menu del MDI frame per rappresentare la MDI child window correntemente attiva.

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

5

## Modifica CChildFrame

- ❖ Proviamo a dividere in due la vista
- ❖ Si deve modificare la nostra classe CChildFrame derivata da CMDIChildWnd
- ❖ Aggiungiamo nella classe CChildFrame

```
CSplitterWnd m_SplitWnd;  
bool m_bSplitterCreated;  
CMBView *m_fview;  
CMBGLView *m_glview;
```

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

6

## CSplitterWnd

- ❖ Facciamo override totale della funzione CChildFrame::OnCreateClient

```
BOOL CChildFrame::OnCreateClient(LPCREATESTRUCT lpcs, CCreateContext* pContext)
{
    m_SplitWnd.CreateStatic(this, 1, 2);
    if (!m_SplitWnd.CreateView(0, 0, RUNTIME_CLASS(CMBView),
        CSize(300, 100), pContext)) {
        return FALSE;
    };
    if (!m_SplitWnd.CreateView(0, 1, RUNTIME_CLASS(CMBGLView),
        CSize(100, 100), pContext)) {
        return FALSE;
    }
    m_fview = (CMBView *) m_SplitWnd.GetPane(0, 0);
    m_glview = (CMBGLView *) m_SplitWnd.GetPane(0, 1);
    m_bSplitterCreated=true;
    return true;
}
```

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

7

## CSplitterWnd

- ❖ In questo modo alla creazione di una nuova finestra per il doc invece di fare una sola view e di attaccarcela crea un oggetto di tipo CSplitterWnd e ci attacca due view di due tipi diversi.
- ❖ Adesso adattiamo la seconda finestra ad opengl come abbiamo visto nella scorsa lezione:

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

8

## Aggiungiamo opengl

- ❖ Per comodità trovate tutti i pezzi di codice che servono in un txt chiamato MFCOpengl.txt
  - ❖ Include opengl
  - ❖ Membri nella classe view
    - ❖ OnCreate
    - ❖ OnSize
    - ❖ OnDestroy
    - ❖ OnEraseBkgnd
    - ❖ SetGL
    - ❖ InitGL

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

9

## Aggiungiamo Moebius

- ❖ Copiamo la subdir vcg che contiene un po' di utility (point3, mesh)
- ❖ Copiamo la classe MoebiusStrip
- ❖ Non vi dettaglio troppo la classe MoebiusStrip, perchè è semplicemente una ristrutturazione (espone un po' dei parametri usati) del codice visto nelle lez precedenti.
- ❖ Aggiungiamo nel doc un oggetto di tipo MoebiusStrip.
- ❖ E nella ondraw un check se è vuoto lo ricreo.
- ❖ if(pd->m.Empty()) pd->m.Generate();
- ❖ Attenzione agli errori da inclusioni mancate...

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

10

## Moebius Strip

```
class MoebiusStrip
{
public:
    MoebiusStrip(void);
    ~MoebiusStrip(void);
    void Generate();
    // parameters

    int SideNum; // 2 = strip, 3 triangular section ecc.
    float Turns; // espresso in 2pi / SideNum.
    float Radius; // del centro della strip
    float Width; // larghezza lato strip;
    float Height; // lato "verticale"

    int StepNum;
    CMesh Ring;
    bool Empty() const {return Ring.Empty();}
    void Clear();
    void GenerateRing(const vector<Point3F> &Section);
    void GenerateSection(vector<Point3F> &Section);
};
```

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

11

## Cosa non va in questa classe?

- ❖ Per come è strutturata ora mi viene difficile:
- ❖ Calcolo delle normali hacked
- ❖ fare sezioni qualsiasi
- ❖ Stabilire come si muove un oggetto sopra l'anello
- ❖ Mancano varie modalità di rendering
- ❖ Generazione texture coords
- ❖ Non so cosa voglio...

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

12

## OnIdle

- ❖ Questa volta la onidle deve anche gestire il fatto che non tutte le view che trovo devono essere ridisegnate.
- ❖ Per sapere a runtime di che tipo è un oggetto si può usare I cast del C++
  - ❖ `static_cast<type>(expr)`
  - ❖ `dynamic_cast<type>(expr)`
  - ❖ `reinterpret_cast<type>(expr)`
- ❖ Il `dynamic_cast` puro richiede dati aggiuntivi per ogni classe...

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

13

## OnIdle

```
BOOL CMOebius4App::OnIdle(LONG lCount)
{
    int dcnt = 0;
    POSITION pos = GetFirstDocTemplatePosition();
    while (pos != NULL) {
        CDocTemplate* pt = GetNextDocTemplate(pos);
        POSITION dpos = pt->GetFirstDocPosition();
        while (dpos != NULL) {
            dcnt++;
            CMBDoc *pd=(CMBDoc *)pt->GetNextDoc(dpos);
            POSITION vpos = pd->GetFirstViewPosition();
            while (vpos != NULL) {
                CView *base=pd->GetNextView(vpos);
                if (dynamic_cast<CMBGLView *>(base) ){
                    CMBGLView* pView = (CMBGLView*)base;
                    pView->Invalidate();
                }
            }
        }
    }
    return dcnt;
}
25 Nov 2002
```

Costruzione di Interfacce - Paolo Cignoni

14

## Dynamic Casting a lá mfc

- ❖ Di default nel compilatore runtime type info sono disabilitate (/GR)
- ❖ Alternativa usare una macro mfc
- ❖ `DYNAMIC_DOWNCAST(classtype, pointer)`

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

15

## Aggiungiamo un po' di controlli

- ❖ Nelle risorse cerchiamo il dialog che corrisponde al formview di sinistra
- ❖ E aggiungiamo tre edit control, tre stringhe di testo statiche e un bottone di apply
- ❖ Per ogni edit control aggiungiamo una variabile del tipo `CEdit` nella view corrispondente



25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

16

## Gestione Controlli

- ❖ Scriviamo il gestore dell'evento pressione del tasto apply

```
void CMBView::OnClickedButtonApply()
{
    CString buf;
    CMBDoc *pd=GetDocument();
    m_ctrlRadius.GetWindowText(buf);
    pd->m.Radius=atof(buf);

    m_ctrlWidth.GetWindowText(buf);
    pd->m.Width=atof(buf);

    m_ctrlHeight.GetWindowText(buf);
    pd->m.Height=atof(buf);

    pd->m.Clear();
    pd->UpdateAllViews(0);
}
25 Nov 2002
```

Costruzione di Interfacce - Paolo Cignoni

17

## Gestione Controlli

- ❖ Scriviamo la on initial update per inizializzare i vari edit a valori sensati.

```
void CMBView::OnInitialUpdate()
{
    CFormView::OnInitialUpdate();
    CString buf;
    buf.Format("%f",GetDocument()->m.Width);
    m_ctrlWidth.SetWindowText(buf);
    buf.Format("%f",GetDocument()->m.Height);
    m_ctrlHeight.SetWindowText(buf);
    buf.Format("%f",GetDocument()->m.Width);
    m_ctrlRadius.SetWindowText(buf);

    ResizeParentToFit(false);
}
25 Nov 2002
```

Costruzione di Interfacce - Paolo Cignoni

18

## Accelerator

- ❖ Per far sì che premendo enter si aggiorni tutto si può assegnare come shortcut al tasto apply la pressione del tasto enter
- ❖ Basta aggiungerne tra gli accelerator nelle risorse uno appropriato.

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

19

## Aggiungiamo il texturing

- ❖ Usiamo Devil
  - ❖ Aggiungere path include e lib
  - ❖ Aggiungere nella view opengl
    - ❖ #include <i1/i1.h>
    - ❖ #include <i1/ilu.h>
    - ❖ #include <i1/ilut.h>

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

20

## Init Devil

- ❖ Nella OnInitialUpdate della view opengl si deve mettere l'inizializzazione della Devil

```
❖ SetGL();  
❖ ilInit();  
❖ ilutInit();  
❖ ilutRenderer(ILUT_OPENGL);  
❖ ti=ilutGLLoadImage("image.jpg");  
❖ ilutGLBuildMipmaps();
```

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

21

## Nella OnDraw

```
glEnable(GL_TEXTURE_2D);  
glBindTexture(GL_TEXTURE_2D,ti);  
glTexEnvf(GL_TEXTURE_ENV, GL_TEXTURE_ENV_MODE, GL_DECAL);  
  
glTexParameterf(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_LINEAR_MIPMAP_LINEAR);  
glEnable(GL_TEXTURE_GEN_S);  
glEnable(GL_TEXTURE_GEN_T);  
glTexGeni(GL_S, GL_TEXTURE_GEN_MODE, GL_OBJECT_LINEAR);  
glTexGeni(GL_T, GL_TEXTURE_GEN_MODE, GL_OBJECT_LINEAR);  
float planevs[4]={1.0,0,0,0};  
float planevt[4]={0,1.0,0,0};  
glTexGenfv(GL_S, GL_OBJECT_PLANE, planevs);  
glTexGenfv(GL_T, GL_OBJECT_PLANE, planevt);
```

25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

22

## Nastro mirrored

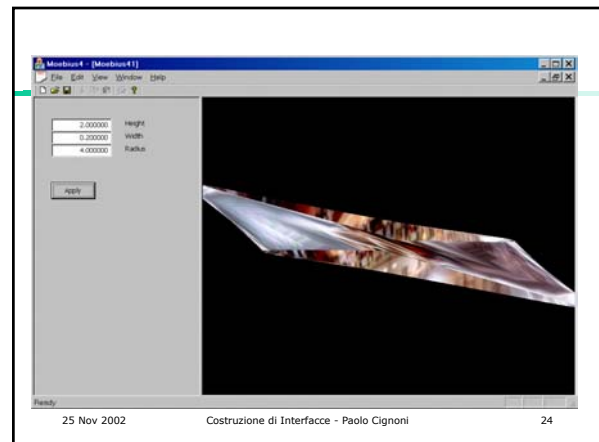
- ❖ Sostituire nella ondraw la parte di generazione di coord texture (le ultime 6 righe) con
  - ❖ glTexGeni(GL\_S, GL\_TEXTURE\_GEN\_MODE, GL\_SPHERE\_MAP);
  - ❖ glTexGeni(GL\_T, GL\_TEXTURE\_GEN\_MODE, GL\_SPHERE\_MAP);
- ❖ E caricare un'immagine sensata per questo tipo di mapping



25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

23



25 Nov 2002

Costruzione di Interfacce - Paolo Cignoni

24

## Conclusioni

---

- ❖ Stiamo ancora sperimentando
- ❖ Mancano gli oggetti che si muovono sull'anello
- ❖ Dobbiamo decidere meglio le specifiche dell'app.
- ❖ L'animazione ora è gestita interamente dentro la view. Non portabile...
- ❖ Butteremo via ancora varie volta quasi tutto...