

Costruzione di Interfacce Lezione 28 MMeditor widgets cubemaps ecc,

cignoni@iei.pi.cnr.it
<http://vcg.isti.cnr.it/~cignoni>

un memento

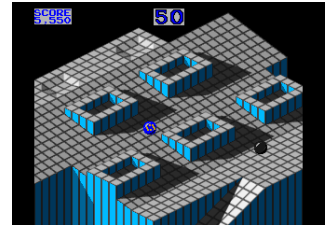
- ❖ Come distribuire I progetti
- ❖ togliete tutto il superfluo
 - ❖ lasciate l'exe
 - ❖ tutti I file inutili fatti dal .net
 - ❖ *.ilk *.ncb *.pdb *.idb *.obj *.suo
- ❖ togliete tutto l'eliminabile
 - ❖ tutti I file del .net che possono essere rifatti dal pro
 - ❖ tutti I file che vengono generati dai precompilatori di qt
 - ❖ quelli generati dal moc a partire da .h che hanno la macro Q_OBJECT
 - ❖ quelli generati dal uic a partire da file *.ui

memento2

- ❖ State attenti a tutte le lib che ci devono essere anche nel .pro
- ❖ non assumete che nel .net ci siano già I path di include e lib.
- ❖ dopo che avete fatto la distrib, provatela.

Continuiamo con l'editor

- ❖ A che cosa miriamo?



Progettiamo un po'

- ❖ Voglio poter
 - ❖ cambiare la geometria del terreno
 - ❖ cambiare le proprietà del terreno
 - ❖ sia di gioco che visuali
 - ❖ aggiungere elementi al gioco sul terreno
 - ❖ Cambiare alcune proprietà globali del livello

Interfaccia dell'editor

- ❖ tre modalità
- ❖ Game
 - ❖ Proprietà globali del gioco
 - ❖ grandezza terreno
 - ❖ sfondo, durata ecc
- ❖ Geometry
 - ❖ forma del terreno
- ❖ Objects
 - ❖ oggetti in gioco
 - ❖ Decorativi e non.

Tab

- ❖ Realizzo le tre porzioni dell'interfaccia con un tab
- ❖ fare attenzione alle sizePolicy del widget CIGLWidget
 - ❖ con quelle di default se si fa un layout orizzontale vince sempre il tab...
 - ❖ cambiarle e cambiare anche lo stretch
 - ❖ 1 al tab
 - ❖ 3 a gl

distribuzione delle funzionalita'

- ❖ Nella toolbar ci andrebbero solo quelle cose che hanno sempre senso
 - ❖ file/save
 - ❖ edit pick vs trackball
 - ❖ rendering modes wireframe /flat

Primo tab: Cose Globali

- ❖ Nome del livello
- ❖ Scelta della cubemap
 - ❖ dare almeno un paio di possibilita' all'utente:
 - ❖ scrivere direttamente il nome del file
 - ❖ stringa vuota dovrebbe essere valido
 - ❖ pulsante di browsing
- ❖ Grandezza della mappa

Una nota sulle cubemap

- ❖ per usarle come sfondo basta applicare la trasformazione corrente SENZA rotazioni e scaling e disegnare un cubo con sopra la cube map stessa.
- ❖ farlo a partire dalla modelview corrente
- ❖ Gestire il caso in cui non sia presente l'ext
 - ❖ si disegna un cubo con sei texture
 - ❖ si notano i bordi...

Pure rotation matrix

- ❖ Per ottenere a partire dalla modelview corrente una matrice di rotazione pura basta cancellare la traslazione e togliere lo scaling...

```
glGetModelViewMatrix(mm);  
mm.a[0][3]=0;  
mm.a[1][3]=0;  
mm.a[2][3]=0;  
mm.a[3][3]=1;  
float dt=1.0f/pow(mm.Det(),1.0/3.0);  
Matrix44f ss; ss.Scale(dt,dt,dt);  
rr=mm*ss;
```

Cambiamenti allo SG

- ❖ Cambiato i nomi delle funz per unif con qt (da glDraw a drawGL)
- ❖ aggiunto una funzione initializeGL
 - ❖ per ogni nodo fa quelle cose che necessitano di un contesto gl
 - ❖ loading texture
 - ❖ creazione DL

CIMsg::checkGLError

- ❖ debugging di opengl difficile
- ❖ uno dei modi e' controllare che le varie funzioni opengl non restituiscano errore
 - ❖ `glGetError` restituisce un codice numerico
 - ❖ si riferisce all'ultimo errore (e non all'ultima istruzione!)
 - ❖ quindi si deve andare all'indietro fino a trovare un comando gl in cui si sia sicuri che prima non c'era errore e dopo si'...
- ❖ classettina `CIMsg::checkGLError(err,title);`

Ma dove teniamo i dati?

- ❖ Sempre nel nostro scenegraph...
- ❖ aggiungiamo il nodo per la cubemap di sfondo
 - ❖ questa volta la classe wrapper del nodo tiene qualcosa di piu'
 - ❖ nome base della texture
 - ❖ matrice di rotazione da applicare

Editing

- ❖ Concetto di selezione
 - ❖ si puo' selezionare un insieme di faccette
 - ❖ tool per la selezione
- ❖ azione sulla selezione
 - ❖ si puo' fare qualcosa
 - ❖ alzarle/abbassarle
 - ❖ farne una rampa
 - ❖ ecc
- ❖ come si tiene la selezione?
 - ❖ due possibilita'
 - ❖ implicita con dati nella cella
 - ❖ esplicita con lista di rif a celle
 - ❖ entrambe le soluzioni hanno vantaggi e svantaggi...

edit/track

- ❖ Aggiungiamo due azioni sulla toolbar per passare di modo tra editing e trackball
- ❖ si fa un action group
 - ❖ dentro si aggiunge due azioni
 - ❖ si mette che entrambe sono di tipo toggle e il gruppo e' esclusivo
 - ❖ una delle due (la track) la mettiamo ad on alla partenza.

Colleghiamo il tutto

- ❖ mettiamo una var nel widgetgl che dice in che stato siamo
 - ❖ `bool Editing=false;`
- ❖ aggiungiamo uno slot `setEditing(bool)` alla widget
- ❖ e comunichiamo esplicitamente il fatto al designer...
- ❖ facciamo la connessione tra il segnale toggle dell'action edit e lo slot `setEditing` della `glView`

Editing vs trackball

- ❖ basta controllare lo stato nelle tre funzioni del mouse.

```
void CIGLWidget::mousePressEvent(QMouseEvent*e){
    if(!Editing){
        tb.MouseDown( e->x(), e->y(), 0 );
        update();
    }
}
void CIGLWidget::mouseMoveEvent(QMouseEvent*e){
    if(!Editing){
        tb.CalcRotMatrix( rot, e->x(), e->y());
        update();
    }
}
void CIGLWidget::mouseReleaseEvent(QMouseEvent*e){
    if(!Editing){
        tb.MouseUp( e->x(), e->y());
        update();
    }
}
```

Nota

- ❖ se non appaiono le immagini una volta compilato significa che c'è un disallineamento tra pro e sln
- ❖ non sa che ci sono altre immagini da usare per fare la imagecollection.
- ❖ rifare la solution a partire dal pro.