

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion



## Synthèse d'images et animation OpenGL - Cours 2

Estelle Duveau - [estelle.duveau@inria.fr](mailto:estelle.duveau@inria.fr)

3A IRV, 01 octobre 2008

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- **Cours** : introduction à OpenGL - 24/09/2008 (1h30)
- **Cours** : OpenGL avancé - 01/10/2008 (1h30)
- **TP1** : découverte et affichage - 08/10/2008 (1h30)
- **TP2** : découverte et affichage - 15/10/2008 (1h30)
- **TP3** : utilisation des textures - 05/11/2008 (1h30)
- **TP4** : animation de personnages +  
**TP5** : modèles physiques de base - 19/11/2008 (3h)
- **TP6** : modèles physiques avancés - 26/11/2008 (1h30)
- **Projet** - semaine du 1er décembre (2 jours)

Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

## ② Display lists

Plaquage de  
textures

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

Charger une texture  
Créer un texture object  
Indiquer comment appliquer la texture à chaque pixel  
Activer le plaquage  
Attacher la texture à l'objet

Les buffers  
OpenGL

## ④ Les buffers OpenGL

Sélection  
d'objets

## ⑤ Sélection d'objets

Conclusion

## ⑥ Conclusion

## Blending, anti-aliasing et autres effets

- Blending
- Anti-aliasing
- Brouillard
- Paramètres de points

## Display lists

## Plaquage de textures

- Charger une texture
- Créer un texture object
- Indiquer comment appliquer la texture à chaque pixel
- Activer le plaquage
- Attacher la texture à l'objet

## Les buffers OpenGL

## Sélection d'objets

## Conclusion

### ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

### ② Display lists

### ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

### ④ Les buffers OpenGL

### ⑤ Sélection d'objets

### ⑥ Conclusion

Blending,  
anti-aliasing et  
autres effets

## Blending

Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

## ② Display lists

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

## ④ Les buffers OpenGL

## ⑤ Sélection d'objets

## ⑥ Conclusion

Blending,  
anti-aliasing et  
autres effets

## Blending

Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- **Blending** = fusion, mélange → de couleurs
- Objectif : obtenir des effets de transparence en combinant les couleurs d'une **source** et d'une **destination**
- Les effets obtenus dépendront de la manière de combiner les couleurs : plusieurs **fonctions de blending** possibles
- Utilisation du **canal alpha (RBGA)**
- Pas possible en mode color-index (il n'a pas d'info de composante alpha)

## Blending : principe

Blending,  
anti-aliasing et  
autres effets

### Blending

Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- Combinaison de la couleur du pixel traité (**source**) avec la valeur du pixel déjà présent dans le framebuffer (**destination**)
- Intervient juste avant l'affichage du framebuffer à l'écran

### Exemple :

Vue d'un **objet** à travers un **verre vert**, qui transmet 80% de la lumière (opacité = 20%) :

couleur à afficher

= 20% couleur du verre et 80% couleur de l'objet

= 20% nouvelle couleur + 80% ancienne couleur

=  $\alpha_{new} RGB A_{new} + (1.0 - \alpha_{new}) RGB A_{old}$

# Blending : en pratique

Blending,  
anti-aliasing et  
autres effets

## Blending

Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- Plus général : couleur =  
 $(a_1 R_{new} + a_2 R_{old}, b_1 G_{new} + b_2 G_{old}, c_1 B_{new} + c_2 B_{old}, d_1 \alpha_{new} + d_2 \alpha_{old})$
- 2 commandes possibles :
  - `glBlendFunc(srcFactor, destFactor)`
  - `glBlendFuncSeparate(srcRGB, destRGB, srcAlpha, destAlpha)`
- Possibilités pour les paramètres : `GL_ZERO`, `GL_ONE`, `GL_SRC_ALPHA`, `GL_ONE_MINUS_SRC_ALPHA`, `GL_CONSTANT_COLOR`, `GL_CONSTANT_ALPHA`,  
...
- Ne pas oublier `glEnable(GL_BLEND)` !



## Blending : en pratique

Blending,  
anti-aliasing et  
autres effets

## Blending

Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
texturesCharger une  
textureCréer un texture  
objectIndiquer  
comment  
appliquer la  
texture à chaque  
pixelActiver le  
plaquage  
Attacher la  
texture à l'objetLes buffers  
OpenGLSélection  
d'objets

Conclusion

Paramètre	Facteur de blending
<code>GL_ZERO</code>	$(0.0, 0.0, 0.0, 0.0)$
<code>GL_ONE</code>	$(1.0, 1.0, 1.0, 1.0)$
<code>GL_SRC_ALPHA</code>	$(\alpha_{source}, \alpha_{source}, \alpha_{source}, \alpha_{source})$
<code>GL_ONE_MINUS_SRC_ALPHA</code>	$(1, 1, 1, 1) - (\alpha_{source}, \alpha_{source}, \alpha_{source}, \alpha_{source})$
<code>GL_CONSTANT_COLOR</code>	$(R_{constant}, G_{constant}, B_{constant}, \alpha_{constant})$
<code>GL_CONSTANT_ALPHA</code>	$(\alpha_{constant}, \alpha_{constant}, \alpha_{constant}, \alpha_{constant})$
...	...

Dans les deux derniers cas, spécifier la constante avec  
`glBlendColor(cstRed, cstGreen, cstBlue, cstAlpha)`

# Fonctions de blending

Blending,  
anti-aliasing et  
autres effets

## Blending

Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- On peut faire plus compliqué qu'une simple addition de coefficients :
  - `GL_FUNC_ADD` :  $k_1 RGBA_{new} + k_2 RGBA_{old}$
  - `GL_FUNC_SUBTRACT` :  $k_1 RGBA_{new} - k_2 RGBA_{old}$
  - `GL_FUNC_REVERSE_SUBTRACT` :  $k_2 RGBA_{old} - k_1 RGBA_{new}$
  - `GL_MIN` :  $\min(k_1 RGBA_{new}, k_2 RGBA_{old})$
  - `GL_MAX` :  $\max(k_1 RGBA_{new}, k_2 RGBA_{old})$
  - `GL_LOGIC_OP` :  $RGBA_{new} \text{ op } RGBA_{old}$
- A utiliser dans `glBlendEquation(mode)` ou `glBlendEquationSeparate(modeRGB, modeAlpha)`

# Exemples d'utilisation du blending

Blending,  
anti-aliasing et  
autres effets

## Blending

Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer une texture  
objet  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

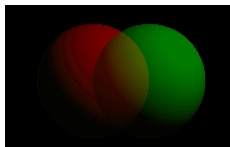
Sélection  
d'objets

Conclusion

## Exemple :

Mélange de deux images à proportions égales

- 1 `glEnable(GL_BLEND);`
- 2 `glBlendFunc(GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA);`
- 3 `glMaterialfv(GL_FRONT_AND_BACK, GL_DIFFUSE, (1, 0, 0, 0.5));`
- 4 `drawSphere();`
- 5 `glMaterialfv(GL_FRONT_AND_BACK, GL_DIFFUSE, (0, 1, 0, 0.5));`
- 6 `drawSphere();`



Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

Blending  
**Anti-aliasing**  
Brouillard  
Paramètres de points

Display lists

## ② Display lists

Plaquage de textures

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

Charger une texture  
Créer un texture object  
Indiquer comment appliquer la texture à chaque pixel  
Activer le plaquage  
Attacher la texture à l'objet

Les buffers  
OpenGL

## ④ Les buffers OpenGL

Sélection  
d'objets

## ⑤ Sélection d'objets

Conclusion

## ⑥ Conclusion

# Antialiasing

Blending,  
anti-aliasing et  
autres effets

Blending

**Anti-aliasing**

Brouillard

Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture

Créer un texture  
object

Indiquer  
comment  
appliquer la  
texture à chaque  
pixel

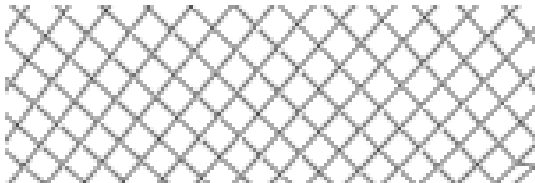
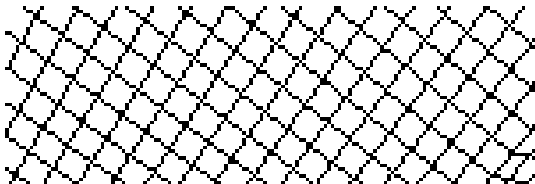
Activer le  
plaquage

Attacher la  
texture à l'objet

Les buffers  
OpenGL

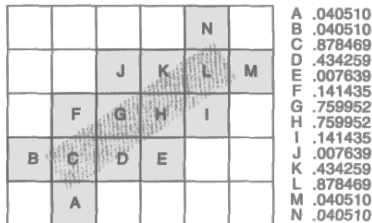
Sélection  
d'objets

Conclusion



## Antialiasing : principe

- Calcul d'une valeur  $c$  de **couverture** pour chaque pixel traversé par la ligne  
→  $c =$  **pourcentage de l'aire du pixel traversé par la ligne**
- Multiplication de la valeur de la composante  $\alpha$  par  $c$
- Blend du pixel avec la couleur précédemment stockée dans le framebuffer



Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

# Antialiasing : en pratique

Blending,  
anti-aliasing et  
autres effets

Blending  
**Anti-aliasing**  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- **Problème** : le calcul des valeurs de couverture n'est pas simple
- Quelques solutions :
  - ① utiliser `glEnable(GL_LINE_SMOOTH)` ou `glEnable(GL_POINT_SMOOTH)`  
⇒ aucun contrôle sur la qualité du résultat
  - ② ajouter `glBlendFunc(GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA)` et éventuellement jouer sur `glLineWidth(myWidth)` et `glPointSize(mySize)`
  - ③ faire du **multisampling** (décomposition des calculs pour un pixel en calculs plus simples pour des morceaux de pixels)
- D'autres existent (utilisation de textures, du buffer d'accumulation, ...)

Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

Blending  
Anti-aliasing

**Brouillard**  
Paramètres de  
points

Display lists

## ② Display lists

Plaquage de  
textures

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

Charger une  
texture  
Créer un texture  
object

Indiquer  
comment  
appliquer la  
texture à chaque  
pixel

Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

## ④ Les buffers OpenGL

Sélection  
d'objets

## ⑤ Sélection d'objets

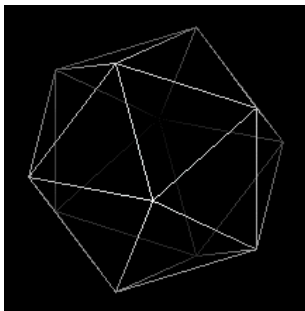
Conclusion

## ⑥ Conclusion



## Brouillard (*fog*) : principe

- Plus un objet est loin de la caméra, plus sa couleur prend celle du brouillard
- Permet de simuler le brouillard, mais aussi la fumée, la pollution, la vapeur, la buée, ...
- **Depth cu(e)ing** (points et lignes) : réduction de l'intensité d'un objet en fonction de la distance à l'observateur  
→ même principe



Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
**Brouillard**  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

## Brouillard (*fog*) : réglages

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
**Brouillard**  
Paramètres de  
paramètres

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- Mélange de la couleur du brouillard et de la couleur de l'objet selon un **facteur**  $f$  :

$$RGBA = fRGBA_{objet} + (1 - f)RGBA_{fog}$$

- $f$  dépend de la **profondeur**  $z$  de l'objet dans la scène et peut prendre trois formes :

$$f = e^{-density \cdot z} \quad GL\_EXP$$

$$f = e^{-(density \cdot z)^2} \quad GL\_EXP2$$

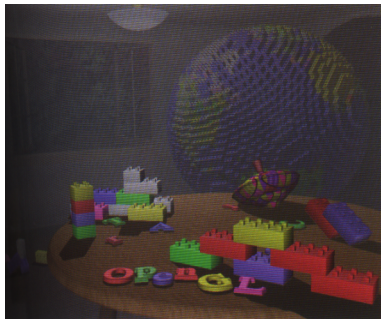
$$f = \frac{end - z}{end - start} \quad GL\_LINEAR$$

- $density$ ,  $start$  et  $end$  à fixer dans les variables **GL\_FOG\_DENSITY**, **GL\_FOG\_START** et **GL\_FOG\_END**
- Aussi possible en mode Color Index
- On peut aussi changer la direction de  $z$  avec **glFogCoord(...)**

## Brouillard (*fog*) : en pratique

### Exemple :

```
glEnable(GL_FOG);  
GLfloat fogColor[4] = {0.5, 0.5, 0.5, 1.0 };  
glFogi(GL_FOG_MODE, GL_EXP);  
glFogfv(GL_FOG_COLOR, fogColor);  
glFogf(GL_FOG_DENSITY, 0.35);
```



Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
**Brouillard**  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

## ② Display lists

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

## ④ Les buffers OpenGL

## ⑤ Sélection d'objets

## ⑥ Conclusion

## Paramètres de points

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard

Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- **Objectif** : afficher des points de différentes tailles (par ex. pour simuler des gouttes de pluie)
- `glPointSize()` est une première solution mais peu de contrôle
- `glPointParameterf(pname, param)` permet d'avoir plus de contrôle
- `pname` : `GL_POINT_DISTANCE_ATTENUATION`, `GL_POINT_SIZE_MIN`, `GL_POINT_SIZE_MAX`, ...
- Si `GL_POINT_DISTANCE_ATTENUATION`,

$$taille_{new} = \frac{taille_{old}}{\sqrt{a+b.z+c.z^2}}, (a, b, c) = param$$

Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

Blending  
Anti-aliasing  
Brouillard  
Paramètres de points

Display lists

## ② Display lists

Plaquage de textures

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

Charger une texture  
Créer un texture object  
Indiquer comment appliquer la texture à chaque pixel  
Activer le plaquage  
Attacher la texture à l'objet

Les buffers  
OpenGL

## ④ Les buffers OpenGL

Sélection  
d'objets

## ⑤ Sélection d'objets

Conclusion

## ⑥ Conclusion

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- **Display list** = ensemble de commandes OpenGL stockées pour être exécutées plus tard
- Permet d'**améliorer les performances** : on stocke une seule fois des commandes qui peuvent être exécutées souvent (affichage, changements d'états)  
→ utilisation en réseau : display lists stockées sur le serveur

# Création d'une display list

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- Une liste est identifiée par un entier : son **indice**
- On génère des indices par `glGenLists(range)` qui renvoie le premier indice d'une suite de `range` indices libérés pour identifier des listes
- Une display list comprend toutes les routines OpenGL comprises entre un `glNewList(listIndex, GL_COMPILE)` et `glEndList()`
- `GL_COMPILE` peut être remplacé par `GL_COMPILE_AND_EXECUTE` si on veut que les routines soient exécutées en plus d'être stockées dans la display list



## Gestion d'une display list

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- Exécution avec `glCallList(listIndex)`
- Commandes exécutées dans l'ordre où elles ont été stockées
- Display lists **hiérarchiques** : appel d'une display list dans la création d'une autre
- Pour exécuter  $n$  display lists successivement :  
`glCallLists(n, index_type, ptr_index)`  
→ appeler `glListBase(firstListIndex)` lors de la création des listes pour indiquer l'offset à ajouter à `ptr_index`

## Display lists multiples : exemple

### Exemple :

```
void init() {
    GLuint base;
    base = glGenLists(128);
    glListBase(base);
    glNewList(base+'1', GL_COMPILE);
        drawLetter(data1);
    glEndList();
    ...
    glNewList(base+'Z', GL_COMPILE);
        drawLetter(dataZ);
    glEndList(); }

void printString(GLByte *s) {
    GLint len = strlen(s);
    glCallLists(len, GL_BYTE, s); }
```

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
objet  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

## ② Display lists

Plaquage de  
textures

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

Charger une texture  
Créer un texture object  
Indiquer comment appliquer la texture à chaque pixel  
Activer le plaquage  
Attacher la texture à l'objet

Les buffers  
OpenGL

## ④ Les buffers OpenGL

Sélection  
d'objets

## ⑤ Sélection d'objets

Conclusion

## ⑥ Conclusion

# Plaquage de texture - *Texture mapping*

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

Très général, beaucoup de possibilités :

- **Texture** = tableau 2D de données (couleur, luminance, ...) appelées **texels**
- Plusieurs **formats de texture** possibles (RGBA, profondeur, luminance, intensité)
- Plusieurs **modes de plaquages** possibles (remplacer, mélanger, moduler)
- **Rotations/translations** avant plaquage possibles
- Sur ou sous-échantillonnage
- Une texture peut se **répéter** ou non sur l'objet
- ...

## Plaquage de textures : processus

- 1 Générer ou charger une texture depuis un fichier bitmap
- 2 Créer un *texture object* et lui spécifier une texture
- 3 Indiquer comment la texture doit être appliquée à chaque pixel :
  - Remplacer la couleur par celle de la texture,
  - Mélanger la couleur avec celle de la texture,
  - Moduler la couleur selon celle de la texture.
- 4 Activer le plaquage de textures :

`glEnable(GL_TEXTURE_2D)`  
(ou `GL_TEXTURE_1D`, `GL_TEXTURE_3D` ou `GL_TEXTURE_CUBE_MAP`)
- 5 Dessiner la scène, en indiquant comment la texture va “s’attacher” à l’objet  
→ **coordonnées géométriques et coordonnées de texture**

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l’objet

Les buffers  
OpenGL

Sélection  
d’objets

Conclusion

Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

Blending  
Anti-aliasing  
Brouillard  
Paramètres de points

Display lists

## ② Display lists

Plaquage de textures

## ③ Plaquage de textures

Charger une texture

Charger une texture

Créer un texture object

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Activer le plaquage

Attacher la texture à l'objet

Attacher la texture à l'objet

Les buffers OpenGL

## ④ Les buffers OpenGL

Sélection d'objets

## ⑤ Sélection d'objets

Conclusion

## ⑥ Conclusion

## Chargement d'une texture

- `glTexImage2D(target, level, internalFormat, width, height, border, format, type, ptr_texels)`
- `target` : en général, `GL_TEXTURE_2D`
- `level` : si on fait une texture multirésolution (**mipmap**)
- `internalFormat` : nombre de composantes sélectionnées pour les texels
- `border` : 1 ou 0 suivant si la texture a un bord ou pas
- `format` : composants sélectionnés pour le mapping (RGBA, profondeur, luminance, intensité)
- On peut aussi lire la texture du framebuffer :  
`glCopyTexImage2D(target, level, internalFormat, x, y, width, height, border)`  
((`x, y`) coin bas gauche du rectangle définissant la texture)
- Textures 1D et 3D :  
`glTexImage1D(...)`, `glTexImage3D(...)`

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture

Créer un texture  
object

Indiquer  
comment  
appliquer la  
texture à chaque  
pixel

Activer le  
plaquage

Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

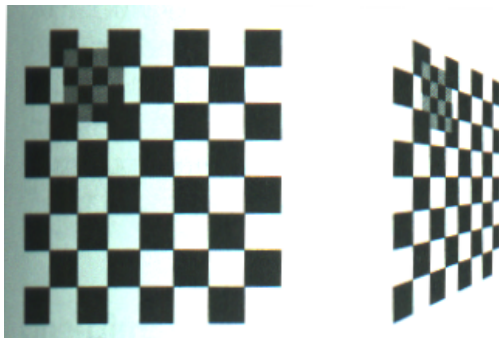
Conclusion

## Remplacement d'une partie de la texture

On peut remplacer une partie de la texture par une autre texture :

```
glTexSubImage2D(target, level, xOffset, yOffset,  
width, height, format, type, ptr_texels)
```

→ évite de créer complètement une nouvelle texture



Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture

Créer un texture  
object

Indiquer  
comment  
appliquer la  
texture à chaque  
pixel

Activer le  
plaquage

Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion



Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

## ② Display lists

Plaquage de  
textures

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

Charger une texture  
Créer un texture object  
Indiquer comment appliquer la texture à chaque pixel  
Activer le plaquage  
Attacher la texture à l'objet

Les buffers  
OpenGL

## ④ Les buffers OpenGL

Sélection  
d'objets

## ⑤ Sélection d'objets

Conclusion

## ⑥ Conclusion

## Création d'une *texture object*

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture

Créer un *texture object*

Indiquer  
comment  
appliquer la  
texture à chaque  
pixel

Activer le  
plaquage

Attacher la  
texture à l'objet

Les buffers

OpenGL

Sélection  
d'objets

Conclusion

- Un *texture object* évite de devoir recharger une image de texture plusieurs fois : il stocke les données d'une texture en mémoire

- Associer un nom (un entier) à  $n$  texture objects :

```
glGenTextures(n, ptr_texNames)
```

- Créer (*bind* = attacher à un nom) et utiliser un *texture object* :

```
glBindTexture(GL_TEXTURE_2D, texName)
```

- Supprimer  $n$  *textures objects* :

```
glDeleteTextures(n, ptr_texNames)
```

Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de points

Blending  
Anti-aliasing  
Brouillard  
Paramètres de points

Display lists

## ② Display lists

Plaquage de textures

## ③ Plaquage de textures

Charger une texture  
Créer un texture object  
Indiquer comment appliquer la texture à chaque pixel  
Activer le plaquage  
Attacher la texture à l'objet

Charger une texture  
Créer un texture object  
Indiquer comment appliquer la texture à chaque pixel  
Activer le plaquage  
Attacher la texture à l'objet

Les buffers  
OpenGL

## ④ Les buffers OpenGL

Sélection  
d'objets

## ⑤ Sélection d'objets

Conclusion

## ⑥ Conclusion

## Filtrage de textures

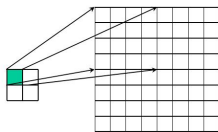
Comment appliquer la texture à chaque pixel :

- Si un pixel correspond à une portion d'un texel,  
**magnification** (sur-échantillonnage)

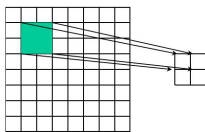
⇒ `glTexParameteri(GL_TEXTURE_2D,  
GL_TEXTURE_MAG_FILTER, GL_NEAREST)`

- Si un pixel correspond à plusieurs texels,  
**minification** (sous-échantillonnage)

⇒ `glTexParameteri(GL_TEXTURE_2D,  
GL_TEXTURE_MIN_FILTER, GL_LINEAR)`



Texture Polygon  
Magnification



Texture Polygon  
Minification

## Fonctions de texture

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- C'était pour **remplacer** la couleur d'un pixel par celle de la texture
- On peut aussi utiliser la texture pour **moduler** la couleur, ou **mélanger** la couleur et celle de la texture
- Fonction de texture : `glTexEnv(target, pname, param)`
- Général et compliqué : voir dans la doc pour les détails
- Une utilisation simple : le mélange
  - `glTexEnvf(GL_TEXTURE_ENV, GL_TEXTURE_ENV_MODE, param)`
  - `param = GL_DECAL, GL_REPLACE, GL_MODULATE, GL_BLEND, GL_ADD` ou `GL_COMBINE`

Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

## ② Display lists

Plaquage de  
textures

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

Charger une texture  
Créer un texture object  
Indiquer comment appliquer la texture à chaque pixel  
**Activer le plaquage**  
Attacher la texture à l'objet

Les buffers  
OpenGL

## ④ Les buffers OpenGL

Sélection  
d'objets

## ⑤ Sélection d'objets

Conclusion

## ⑥ Conclusion

## Génération de textures : exemple

## Exemple :

```

GLubyte Texture[16] =
{
    0,0,0,0, 0xFF,0xFF,0xFF,0xFF,           //Image (2x2)
    0xFF,0xFF,0xFF,0xFF, 0,0,0,0
};

GLuint Nom;

void InitGL()
{
    glClearColor(.5,.5,.5,0);                //Change la couleur du fond
    glEnable(GL_DEPTH_TEST);                //Active le depth test
    glEnable(GL_TEXTURE_2D);                //Active le texturing
    glGenTextures(1,&Nom);                   //Génère un n° de texture
    glBindTexture(GL_TEXTURE_2D,Nom);       //Sélectionne ce n°
    glTexImage2D (
        GL_TEXTURE_2D,                       //Type : texture 2D
        0,                                    //Mipmap : aucun
        4,                                    //Couleurs : 4
        2,                                    //Largeur : 2
        2,                                    //Hauteur : 2
        0,                                    //Largeur du bord : 0
        GL_RGBA,                              //Format : RGBA
        GL_UNSIGNED_BYTE,                    //Type des couleurs
        Texture                               //Adresse de l'image
    );
    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_NEAREST);
    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_NEAREST);
}

```

Blending,  
anti-aliasing et  
autres effetsBlending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
texturesCharger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
**Activer le  
plaquage**  
Attacher la  
texture à l'objetLes buffers  
OpenGLSélection  
d'objets

Conclusion

Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

## ② Display lists

Plaquage de  
textures

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

Charger une texture  
Créer un texture object  
Indiquer comment appliquer la texture à chaque pixel  
Activer le plaquage  
**Attacher la texture à l'objet**

Les buffers  
OpenGL

## ④ Les buffers OpenGL

Sélection  
d'objets

## ⑤ Sélection d'objets

Conclusion

## ⑥ Conclusion



# Coordonnées de texture

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
**Attacher la  
texture à l'objet**

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- Pour chaque sommet de la scène, préciser ses coordonnées  $(s, t)$  de texture  
→ **texture 2D vue comme un carré de largeur et hauteur 1**
- `glTexCoord2f(coords)`

## Coordonnées de texture - Exemple

### Exemple :

```
glBegin(GL_QUADS);  
    glTexCoord2i(0,0);  
    glVertex3i(-1,-1,-1);  
  
    glTexCoord2i(1,0);  
    glVertex3i(+1,-1,-1);  
  
    glTexCoord2i(1,1);  
    glVertex3i(+1,+1,-1);  
  
    glTexCoord2i(0,1);  
    glVertex3i(-1,+1,-1);  
glEnd();
```

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
**Attacher la  
texture à l'objet**

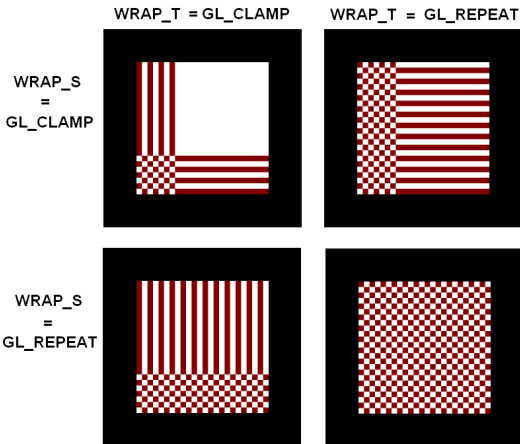
Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

## Répéter la texture

On peut répéter une texture dans une direction avec  
`glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, GL_REPEAT)`



Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
objet  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

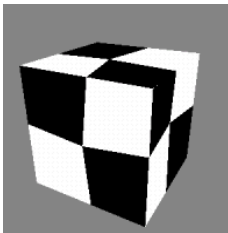
Conclusion

## glHint()

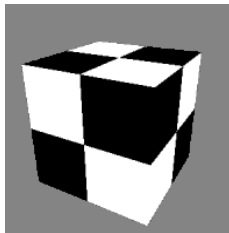
- `glHint(target, hint)` contrôle certains comportements d'OpenGL
- `hint = GL_FASTEST, GL_NICEST` ou `GL_DONT_CARE`
- `target = GL_LINE_SMOOTH_HINT` (antialiasing), `GL_FOG_HINT, GL_TEXTURE_COMPRESSION_HINT, ...`

### Exemple :

`glHint(GL_PERSPECTIVE_CORRECTION_HINT, GL_NICEST)`



Sans



Avec

Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

Blending  
Anti-aliasing  
Brouillard  
Paramètres de points

Display lists

## ② Display lists

Plaquage de textures

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

Charger une texture  
Créer un texture object  
Indiquer comment appliquer la texture à chaque pixel  
Activer le plaquage  
Attacher la texture à l'objet

Les buffers  
OpenGL

## ④ Les buffers OpenGL

## ⑤ Sélection d'objets

Sélection d'objets

## ⑥ Conclusion

Conclusion

# Les buffers OpenGL

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

Quatre buffers forment le **framebuffer** :

- **Color buffers**
  - de 1 à 4 suivant que le système est double bufferisé et qu'il supporte les images stéréoscopiques
  - + éventuellement des buffers auxiliaires
  - `glGetBoolean(GL_DOUBLEBUFFER); glGetBoolean(GL_STEREO);`
- **Depth/Z buffer**
  - ⇒ suppression des parties cachées
- **Stencil buffer**
  - pour ne dessiner qu'une partie de l'image
  - ⇒ simulateur de conduite
- **Accumulation buffer**
  - pour "accumuler" des images
  - ⇒ antialiasing, motion blur

# Utilisation d'un buffer

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

## Initialisation :

- `glClearColor(red, green, blue, alpha)` (RGBA) ou `glClearIndex(red, green, blue, alpha)` (color index)
- `glClearDepth(depth)`
- `glClearStencil(s)` avec `s` un entier
- `glClearAccum(red, green, blue, alpha)`

+ ensuite `glClear(mask)`

avec `mask = GL_COLOR_BUFFER_BIT, GL_DEPTH_BUFFER_BIT, GL_STENCIL_BUFFER_BIT` ou `GL_ACCUM_BUFFER_BIT`

## Sélection d'un color buffer :

- `glDrawBuffer(mode)` ou `glDrawBuffers(n, ptr_buffers)`
- `glReadBuffer(mode)`
- `mode = GL_FRONT, GL_BACK, GL_LEFT, GL_FRONT_LEFT, ...`

# Opérations sur les pixels avec les buffers OpenGL

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- Accepter un pixel selon sa valeur alpha :  
`glAlphaFunc(func, ref)`
- Accepter un pixel selon sa profondeur :  
`glDepthFunc(func)`
- Définir une portion rectangulaire pour le stencil :  
`glScissor(x, y, width, height)`
- Accepter un pixel selon sa valeur par rapport au stencil :  
`glStencilFunc(func, ref, mask)`
- Contrôler l'accumulation buffer :  
`glAccum(op, value)`  
avec `op = GL_ACCUM, GL_LOAD, GL_RETURN, GL_ADD` ou `GL_MULT`



Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

Blending  
Anti-aliasing  
Brouillard  
Paramètres de points

Display lists

## ② Display lists

Plaquage de textures

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

Charger une texture  
Créer un texture object  
Indiquer comment appliquer la texture à chaque pixel  
Activer le plaquage  
Attacher la texture à l'objet

Les buffers  
OpenGL

## ④ Les buffers OpenGL

Sélection  
d'objets

## ⑤ Sélection d'objets

Conclusion

## ⑥ Conclusion

# Sélection d'objets

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- Une scène 3D, c'est encore mieux quand c'est **interactif**
- Deux modes :
  - **Sélection** : l'utilisateur choisit un ou plusieurs objets affichés à l'écran
  - **Feedback** : pour récupérer les infos d'affichage (par exemple pour l'impression ou l'export vers un fichier)
- Dans ces deux modes, les infos d'affichage ne seront pas envoyées au framebuffer : l'affichage est **gelé** tant que l'on est dans un de ces modes

## Pile de noms

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- A chaque primitive sélectionnable est associé un **nom**, dont la valeur est un entier
- Ces entiers sont gérés dans une **pile**
- Initialisation de la pile des noms avec `glInitNames()`
- Association d'un nom à un objet sélectionnable : empiler le nom avec `glPushName(i)`, dessiner l'objet, puis dépiler le nom avec `glPopName(i)`
- Autre possibilité : `glLoadName(i)` qui remplace l'élément en haut de la pile des noms par `i`

## Sélection d'objets : processus

- 1 Spécifier le buffer de sélection :  
`glSelectBuffer(bufMaxSize, bufName)`
- 2 Dessiner la scène
- 3 Passer en mode sélection : `glRenderMode(GL_SELECT)`
- 4 Initialiser la pile des noms : `glInitNames()`
- 5 Définir le volume de visualisation pour la sélection (ne contient pas forcément toute la scène)
- 6 Redessiner la scène et associer son nom à chaque objet sélectionnable
- 7 Sortir du mode sélection et renvoyer les données de sélection :  
`glPopMatrix()`  
`hits = glRenderMode(GL_RENDER)` : nombre de hits

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer une texture  
objet  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

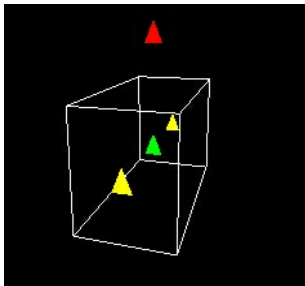
Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- **hit** : une primitive à laquelle est associée un nom intersecte le volume de visualisation
- Hit stocké dans le **buffer de sélection**  
→ on peut sélectionner plusieurs objets à la fois



## Sélection d'objets : exemple

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

### Exemple :

Fonction appelée quand l'utilisateur clique (`glutMouseFunc(func)`) :

```
// Appelé une fois que la scène est dessinée en mode GL_RENDER
glSelectBuffer (BUFSIZE, selectBuf); // Spécifier le buffer de sélection
glRenderMode (GL_SELECT); // Passer en mode sélection
glInitNames(); // Initialiser la pile de noms
glPushName(0); // On va changer de vue
glMatrixMode (GL_PROJECTION); // Définir le volume...
glPushMatrix (); // ...de visualisation...
glLoadIdentity (); //...donc de sélection...
gluPickMatrix (xcursor, ycursor, 5.0, 5.0, viewport);
// ...autour du curseur => fonction GLU
gluOrtho2D (0.0, 3.0, 0.0, 3.0);
glLoadName(1); drawTriangle (); // Nom 1 associé au 1er triangle
glLoadName(2); drawTriangle (); // Nom 2 associé au 2ème triangle
glLoadName(3); drawTriangle (); drawTriangle ();
// Nom 3 associé aux 3ème et 4ème triangles
glMatrixMode (GL_PROJECTION); // On revient à...
glPopMatrix (); // ...la vue initiale
hits = glRenderMode (GL_RENDER); // On récupère les données de sélection
processHits (hits, selectBuf); // On les traite
```

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- Idem sélection : affichage gelé, données envoyées à un buffer
- Le buffer de feedback contient des infos sur les primitives dessinées : type (point, ligne, polygone, ...), sommets, couleur, ... sont stockées sous forme de flottants
- Spécification du buffer avec  
`glFeedbackBuffer(bufSize, bufType, bufname)`  
→ `bufType` dépend du type de primitives affichées :  
`GL_2D`, `GL_3D`, `GL_3D_COLOR`, ...
- Passage en mode feedback avec  
`glRenderMode(GL_FEEDBACK)`
- La taille du buffer est donnée lorsqu'on sort du mode feedback, avec `size = glRenderMode(GL_RENDER)`

Blending,  
anti-aliasing et  
autres effets

## ① Blending, anti-aliasing et autres effets

Blending

Anti-aliasing

Brouillard

Paramètres de points

Blending  
Anti-aliasing  
Brouillard  
Paramètres de points

Display lists

## ② Display lists

Plaquage de textures

## ③ Plaquage de textures

Charger une texture

Créer un texture object

Indiquer comment appliquer la texture à chaque pixel

Activer le plaquage

Attacher la texture à l'objet

Charger une texture  
Créer un texture object  
Indiquer comment appliquer la texture à chaque pixel  
Activer le plaquage  
Attacher la texture à l'objet

Les buffers  
OpenGL

## ④ Les buffers OpenGL

Sélection  
d'objets

## ⑤ Sélection d'objets

Conclusion

## ⑥ Conclusion



Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

# Ce qu'on a vu

- Effets
- Améliorations de performances
- Textures
- Buffers
- Sélection

<http://www-evasion.imag.fr/Membres/Estelle.Duveau/SIA.html>

## Pour en savoir plus

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

### OpenGL :

- Le **red book** :  
<http://www.opengl-redbook.com/>,  
<http://fly.cc.fer.hr/~unreal/theredbook/>
- <http://www.opengl.org/>

### Autour d'OpenGL :

- **GLUT** :  
<http://www.opengl.org/resources/libraries/glut.html>
- **Qt** :  
<http://www.trolltech.com/>
- **libQGLViewer** :  
<http://artis.imag.fr/Members/Gilles.Debonne/QGLViewer/>

# Tutoriaux en ligne

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

- Antoine Bouthors :

<http://evasion.imag.fr/Membres/Antoine.Bouthors/teaching/opengl/>

- Martin Beudet :

<http://eraquila.iquebec.com/site/delphi/opengl/glmenu.htm>

- Nehe :

<http://nehe.gamedev.net/>

- Nate Robins :

<http://www.xmission.com/~nate/tutors.html>

- SGI :

<http://www.sgi.com/products/software/opengl/examples/index.html>

Blending,  
anti-aliasing et  
autres effets

Blending  
Anti-aliasing  
Brouillard  
Paramètres de  
points

Display lists

Plaquage de  
textures

Charger une  
texture  
Créer un texture  
object  
Indiquer  
comment  
appliquer la  
texture à chaque  
pixel  
Activer le  
plaquage  
Attacher la  
texture à l'objet

Les buffers  
OpenGL

Sélection  
d'objets

Conclusion

## La pratique!

Les 2 prochaines séances : TP à rendre

But : Prise en main, modélisation du personnage utilisé tout au long du semestre