

Laboratori de Gràfics, Action plugins.

À. Vinacua, C. Andújar i professors de Gràfics

23 de novembre de 2015

Tipus de plugins



Tipus de plugins

- Effect Plugins

- Canvien l'estat d'OpenGL abans i/o després de que es pinti l'escena.
- Exemples: activar shaders, configurar textures, alpha blending...

- Draw Plugins

- Recórren els objectes per pintar les primitives de l'escena.
- Exemples: dibuixar amb VBO...

- Action Plugins

- Executen accions arbitràries en resposta a events (mouse, teclat).
- Exemples: selecció d'objectes, control de la càmera virtual...

- Render Plugins

- Dibuixar un frame amb un o més passos de rendering.
- Exemples: múltiples passos de rendering, shadow mapping...



Tipus de plugins

- Effect Plugins

- Canvien l'estat d'OpenGL abans i/o després de que es pinti l'escena.
- Exemples: activar shaders, configurar textures, alpha blending...

- Draw Plugins

- Recórren els objectes per pintar les primitives de l'escena.
- Exemples: dibuixar amb VBO...

- Action Plugins

- Executen accions arbitràries en resposta a events (mouse, teclat).
- Exemples: selecció d'objectes, control de la càmera virtual...

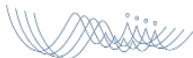
- Render Plugins

- Dibuixar un frame amb un o més passos de rendering.
- Exemples: múltiples passos de rendering, shadow mapping...



Tipus de plugins

- Effect Plugins
 - Canvien l'estat d'OpenGL abans i/o després de que es pinti l'escena.
 - Exemples: activar shaders, configurar textures, alpha blending...
- Draw Plugins
 - Recórren els objectes per pintar les primitives de l'escena.
 - Exemples: dibuixar amb VBO...
- Action Plugins
 - Executen accions arbitràries en resposta a events (mouse, teclat).
 - Exemples: selecció d'objectes, control de la càmera virtual...
- Render Plugins
 - Dibuir un frame amb un o més passos de rendering.
 - Exemples: múltiples passos de rendering, shadow mapping...



Tipus de plugins

- Effect Plugins
 - Canvien l'estat d'OpenGL abans i/o després de que es pinti l'escena.
 - Exemples: activar shaders, configurar textures, alpha blending...
- Draw Plugins
 - Recórren els objectes per pintar les primitives de l'escena.
 - Exemples: dibuixar amb VBO...
- Action Plugins
 - Executen accions arbitràries en resposta a events (mouse, teclat).
 - Exemples: selecció d'objectes, control de la càmera virtual...
- Render Plugins
 - Dibuixar un frame amb un o més passos de rendering.
 - Exemples: múltiples passos de rendering, shadow mapping...



Aquesta sessió: Action plugins



Action plugins

Mètodes propis

- virtual void keyPressEvent (QKeyEvent *)
- virtual void keyReleaseEvent (QKeyEvent *)
- virtual void mouseMoveEvent (QMouseEvent *)
- virtual void mousePressEvent (QMouseEvent *)
- virtual void mouseReleaseEvent (QMouseEvent *)
- virtual void wheelEvent (QWheelEvent *)

Mètodes/Atributs heredats

- virtual void onPluginLoad()
- virtual void onObjectAdd()
- GLWidget* glwidget(); // dóna accés a l'escena i la càmera
- scene()->objects().size() // num objectes
- camera()->getObs() // pos de l'observador

Action plugins

Mètodes propis

- `virtual void keyPressEvent (QKeyEvent *)`
- `virtual void keyReleaseEvent (QKeyEvent *)`
- `virtual void mouseMoveEvent (QMouseEvent *)`
- `virtual void mousePressEvent (QMouseEvent *)`
- `virtual void mouseReleaseEvent (QMouseEvent *)`
- `virtual void wheelEvent (QWheelEvent *)`

Mètodes/Atributs heredats

- `virtual void onPluginLoad()`
- `virtual void onObjectAdd()`
- `GLWidget* glwidget();` // dóna accés a l'escena i la càmera
- `scene()->objects().size()` // num objectes
- `camera()->getObs()` // pos de l'observador

Action plugins

Mètodes propis

- virtual void keyPressEvent (QKeyEvent *)
- virtual void keyReleaseEvent (QKeyEvent *)
- virtual void mouseMoveEvent (QMouseEvent *)
- virtual void mousePressEvent (QMouseEvent *)
- virtual void mouseReleaseEvent (QMouseEvent *)
- virtual void wheelEvent (QWheelEvent *)

Mètodes/Atributs heredats

- virtual void onPluginLoad()
- virtual void onObjectAdd()
- GLWidget* glwidget(); // dóna accés a l'escena i la càmera
- scene()->objects().size() // num objectes
- camera()->getObs() // pos de l'observador

Action plugins

Mètodes propis

- virtual void keyPressEvent (QKeyEvent *)
- virtual void keyReleaseEvent (QKeyEvent *)
- virtual void mouseMoveEvent (QMouseEvent *)
- virtual void mousePressEvent (QMouseEvent *)
- virtual void mouseReleaseEvent (QMouseEvent *)
- virtual void wheelEvent (QWheelEvent *)

Mètodes/Atributs heredats

- virtual void onPluginLoad()
- virtual void onObjectAdd()
- GLWidget* glwidget(); // dóna accés a l'escena i la càmera
- scene()->objects().size() // num objectes
- camera()->getObs() // pos de l'observador

Action plugins

Mètodes propis

- `virtual void keyPressEvent (QKeyEvent *)`
- `virtual void keyReleaseEvent (QKeyEvent *)`
- `virtual void mouseMoveEvent (QMouseEvent *)`
- `virtual void mousePressEvent (QMouseEvent *)`
- `virtual void mouseReleaseEvent (QMouseEvent *)`
- `virtual void wheelEvent (QWheelEvent *)`

Mètodes/Atributs heredats

- `virtual void onPluginLoad()`
- `virtual void onObjectAdd()`
- `GLWidget* glwidget();` // dóna accés a l'escena i la càmera
- `scene()->objects().size()` // num objectes
- `camera()->getObs()` // pos de l'observador

Action plugins

Mètodes propis

- `virtual void keyPressEvent (QKeyEvent *)`
- `virtual void keyReleaseEvent (QKeyEvent *)`
- `virtual void mouseMoveEvent (QMouseEvent *)`
- `virtual void mousePressEvent (QMouseEvent *)`
- `virtual void mouseReleaseEvent (QMouseEvent *)`
- `virtual void wheelEvent (QWheelEvent *)`

Mètodes/Atributs heredats

- `virtual void onPluginLoad()`
- `virtual void onObjectAdd()`
- `GLWidget* glwidget();` // dóna accés a l'escena i la càmera
- `scene()->objects().size()` // num objectes
- `camera()->getObs()` // pos de l'observador

Action plugins

Mètodes propis

- virtual void keyPressEvent (QKeyEvent *)
- virtual void keyReleaseEvent (QKeyEvent *)
- virtual void mouseMoveEvent (QMouseEvent *)
- virtual void mousePressEvent (QMouseEvent *)
- virtual void mouseReleaseEvent (QMouseEvent *)
- virtual void wheelEvent (QWheelEvent *)

Mètodes/Atributs heredats

- virtual void onPluginLoad()
- virtual void onObjectAdd()
- GLWidget* glwidget(); // dóna accés a l'escena i la càmera
- scene()->objects().size() // num objectes
- camera()->getObs() // pos de l'observador

Action plugins

Mètodes propis

- virtual void keyPressEvent (QKeyEvent *)
- virtual void keyReleaseEvent (QKeyEvent *)
- virtual void mouseMoveEvent (QMouseEvent *)
- virtual void mousePressEvent (QMouseEvent *)
- virtual void mouseReleaseEvent (QMouseEvent *)
- virtual void wheelEvent (QWheelEvent *)

Mètodes/Atributs heredats

- virtual void onPluginLoad()
- virtual void onObjectAdd()
- GLWidget* glwidget(); // dóna accés a l'escena i la càmera
- scene()->objects().size() // num objectes
- camera()->getObs() // pos de l'observador

Action plugins

Mètodes propis

- virtual void keyPressEvent (QKeyEvent *)
- virtual void keyReleaseEvent (QKeyEvent *)
- virtual void mouseMoveEvent (QMouseEvent *)
- virtual void mousePressEvent (QMouseEvent *)
- virtual void mouseReleaseEvent (QMouseEvent *)
- virtual void wheelEvent (QWheelEvent *)

Mètodes/Atributs heredats

- virtual void onPluginLoad()
- virtual void onObjectAdd()
- GLWidget* glwidget(); // dóna accés a l'escena i la càmera
- scene()->objects().size() // num objectes
- camera()->getObs() // pos de l'observador

Action plugins

Mètodes propis

- virtual void keyPressEvent (QKeyEvent *)
- virtual void keyReleaseEvent (QKeyEvent *)
- virtual void mouseMoveEvent (QMouseEvent *)
- virtual void mousePressEvent (QMouseEvent *)
- virtual void mouseReleaseEvent (QMouseEvent *)
- virtual void wheelEvent (QWheelEvent *)

Mètodes/Atributs heredats

- virtual void onPluginLoad()
- virtual void onObjectAdd()
- GLWidget* glwidget(); // dóna accés a l'escena i la càmera
- scene()->objects().size() // num objectes
- camera()->getObs() // pos de l'observador

Action plugins

Mètodes propis

- virtual void keyPressEvent (QKeyEvent *)
- virtual void keyReleaseEvent (QKeyEvent *)
- virtual void mouseMoveEvent (QMouseEvent *)
- virtual void mousePressEvent (QMouseEvent *)
- virtual void mouseReleaseEvent (QMouseEvent *)
- virtual void wheelEvent (QWheelEvent *)

Mètodes/Atributs heredats

- virtual void onPluginLoad()
- virtual void onObjectAdd()
- GLWidget* glwidget(); // dóna accés a l'escena i la càmera
- scene()->objects().size() // num objectes
- camera()->getObs() // pos de l'observador

Fluxe de control

Per cada refresc:

- Si hi ha plugins registrats es crida el mètode `preFrame()` de cadascun.
- Si hi ha plugins registrats es crida el mètode `postFrame()` de cadascun.

Tractament d'esdeveniments

Es propaguen als plugins que hi hagi registrats:

```
1 ...  
2 void GLWidget::mousePressEvent( QMouseEvent *e)  
3 {  
4     for (unsigned int i=0; i<plugins.size(); ++i)  
5         qobject_cast<BasicPlugin*>  
6             (plugins[i]->instance())->mousePressEvent(e);  
7 }  
8 ...
```

