ModelViewerl

- + virtual void Initialize()=0
- + virtual void RenderFrame()=0
- + virtual void Quit()=0
- + virtual void AddModel (ModelInfo model info)=0
- + virtual void AddModel (PresetModelInfo model _info)=0
- + virtual void ResetModels()=0
- + virtual bool IsRunning()=0
- + virtual std::mutex & GetMutex()=0

ModelViewer

- std::mutex model_viewer _mutex
- bool should_quit
- bool is_running
- OgreBites::ApplicationContext ctx
- Ogre::Root * ogreRoot
- Ogre::SceneManager* scnMgr
- Ogre::RTShader::ShaderGenerator* shadergen
- Ogre::Light * sceneLight
- Ogre::SceneNode * sceneLightNode
- Ogre::Camera * sceneCamera and 7 more...
- + ModelViewer()
- + bool IsRunning() override
- + void AddModel(ModelInfo model info) override
- + void AddModel(PresetModel Info model info) override
- + void ResetModels() override
- void Initialize() override
- void RenderFrame() override
- void Quit() override
- void Deinitialize()
- std::mutex & GetMutex() override
- void HandleAddModelQueue()
- void HandleResetModelsFlag()

-m -keyHandler

ModelViewer::ModelViewer KeyHandler

- + void addModelViewerContext (ModelViewer *m)
- + bool keyPressed(const OgreBites::KeyboardEvent &evt) override
- + bool mouseWheelRolled (const OgreBites::MouseWheel Event &evt) override
- + bool mouseMoved(const OgreBites::MouseMotionEvent &evt) override
- + bool mousePressed(const OgreBites::MouseButtonEvent &evt) override
- + bool mouseReleased (const OgreBites::MouseButton Event &evt) override

OgreBites::InputListener