

Escola Politècnica Superior
Universitat de Girona

Copy&Paste façanes procedurals per skylineEngine

PROJECTE/TREBALL FI DE CARRERA

Enginyeria Superior en Informàtica. Pla 1997



Document: Resum

Autor: Alejandro Arangua Rovira

Director: Gustavo Ariel Patow

Departament: Informàtica, Matemàtica Aplicada i Estadística

Àrea: LSI

Convocatoria: Setembre 2013

1 Introducció

Avui en dia el modelatge de ciutats (actuals i històriques) és un problema obert pel que no existeixen solucions estàndards ni de codi lliure. Aquest és un problema força important per indústries com la del cinema, la realitat virtual i els videojocs, que sovint necessiten crear grans escenaris. Degut a això, cada vegada es fan servir més eines informàtiques per a la creació d'aquests entorns urbans.



Figura 1: Exemple de modelat de ciutat 3D.

Segurament tothom ha tingut l'oportunitat de veure algun videojoc (com per exemple Assasins Creed o GTA IV) o alguna pel·lícula (com per exemple Cars 2) on l'escenari escollit és una gran ciutat. En el cas de l'Assasins Creed o el GTA IV, els escenaris han estat creats manualment, és a dir, muntant els edificis un per un fins aconseguir una gran ciutat. Això és una feina que ha requerit milers d'artistes i una despesa de temps i de diners més que considerable.

El modelatge procedural pretén solucionar gran part d'aquest problema utilitzant algorismes per tal de generar de forma automàtica les ciutats o edificis que es necessitin. El procés consisteix en deixar que l'ordinador faci la feina complicada i d'aquesta forma l'usuari simplement ha d'especificar els diferents paràmetres per tal d'aconseguir el resultat desitjat.

2 Objectius

L'objectiu d'aquest projecte és el desenvolupament d'una eina d'alt nivell pel modelatge d'edificis procedurals que permeti copiar i enganxar parts arbitràries d'un edifici en un altre. Els edificis procedurals es basen en l'execució iterativa d'un conjunt de regles, que es poden representar per un graf d'operacions. Per tant, l'operació de copiar i enganxar es centra en la reescriptura dels grafs de regles amb l'objectiu de modificar els edificis per tal de dur aquesta tasca. Donat que es treballa sobre la plataforma de recerca anomenada **skylineEngine**, que s'executa sobre el programari Houdini 3D, l'aplicació també estarà implementada a sobre d'aquesta plataforma.

Aquesta eina és un gran avanç respecte les eines que existeixen pels dissenyadors, ja que encapsula un gran ventall d'operacions de re-escriptura de grafs de forma senzilla i transparent per l'usuari.

Un objectiu secundari és integrar aquest mecanisme a dins del **buildingEngine**, que és part de **skylineEngine**, una llibreria pel desenvolupament procedural d'edificis i ciutats desenvolupada pel Grup de Geometria i Gràfics de la UdG.

3 Explicació de la problemàtica

Actualment, cada cop que l'usuari vol replicar geometria mitjançant les eines existents, se l'hi planteja un seguit d'accions llargues i tedioses que molts cops condueixen a resultats indesitjats. Moltes plataformes per a la creació de modelats 3D tals com 3DStudio Max, Blender, Maya, Lightwave, Cinema4D, SoftImage, etc, permeten replicar geometria, però ho fan de manera tradicional. Els passos que ha de seguir l'usuari són llargs i feixucs i no sempre donen els resultats finals que l'usuari desitjaria. A més l'usuari ha de tenir coneixements tècnics sobre com s'estructura la informació dins de Houdini i **buildingEngine** que com a artista o modelador no hauria de tenir presents.

Per a realitzar aquest projecte es va agafar de referència l'article Barroso et al. [1]. Aquest article proposa una solució a aquesta problemàtica de manera procedural i transparent per a l'usuari. Es pot veure un exemple de les operacions que es realitzen automàticament per tal de suplir aquesta tasca en la Figura 2

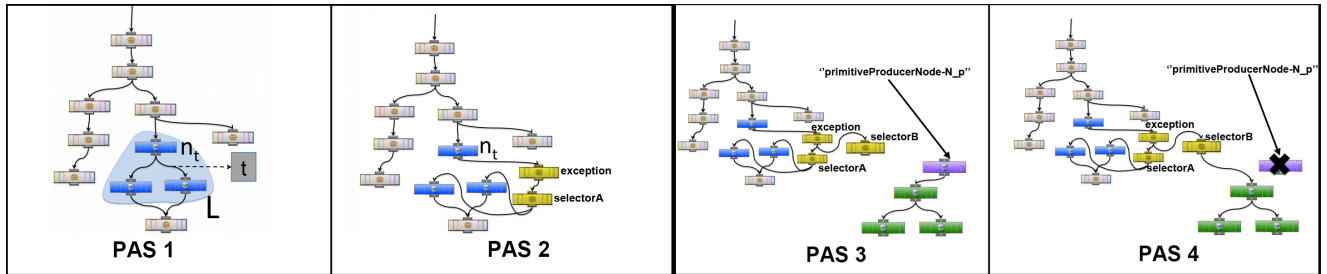


Figura 2: Procés d'enganxat de geometria mitjançant excepcions procedurals.

Es requereix la construcció d'una eina que permeti abstraure a l'usuari de tot aquesta complicació tècnica i que procuri uns resultats més ràpids i precisos.

A més d'aquesta problemàtica, quan es replica geometria en una escena 3D, es fa més densa en polígons i això fa que el nivell d'interactivitat descendeixi ja que la tarja gràfica ha de realitzar un major nombre de càlculs. Per a solucionar això, s'ha de dotar d'alguna eina a l'usuari que permeti realitzar operacions amb una major velocitat d'interacció.

4 Disseny

Cal dir que es va plantejar el sistema des del punt de vista de l'usuari, ja que el sistema havia de ser tan còmode i usable com fos possible.

4.1 Diagrames de cas d'ús generals.

En aquest article proposa una solució a una eina per a realitzar replicació de geometria mitjançant modelatge procedural, i acota de quina manera es copien i re-escriuen les regles, mitjançant excepcions, per a poder dur a terme aquesta tasca.

En la Figura 3 es pot veure el diagrama de cas d'ús principal del projecte.

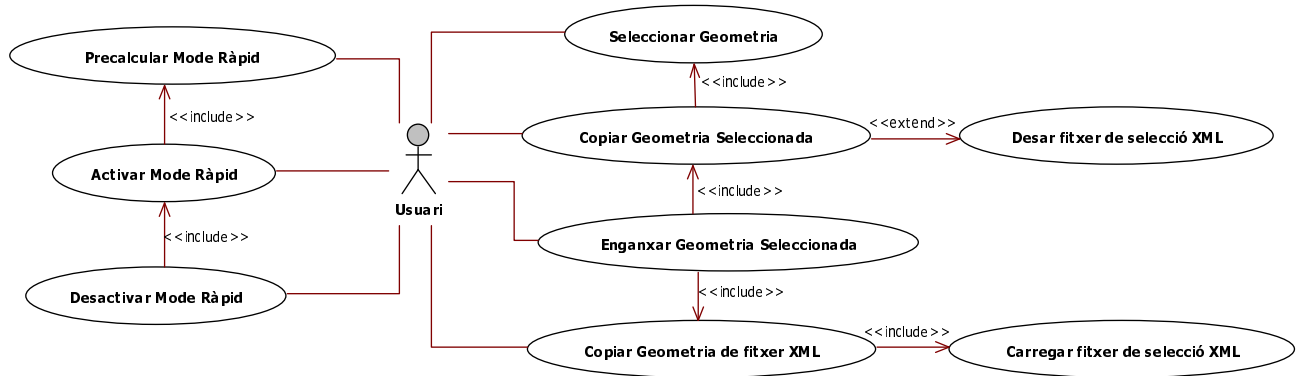


Figura 3: Diagrama de cas d'ús de context general d'acceleració.

4.2 Diagrama de classes

Com es pot veure en la Figura 4 del diagrama de classes complet, només s’han esmentat els mètodes implementats per a aquest projecte de manera que es pugui distingir entre les classes i funcionalitats existents i les pròpies d’aquest projecte. Les classes que no contenen mètodes són classes realitzades en altres projectes i que ara són part del conjunt de classes de **buildingEngine**. Tot i això, s’hi han afegit al diagrama ja que són importants en el desenvolupament de les funcionalitats creades en aquest projecte.

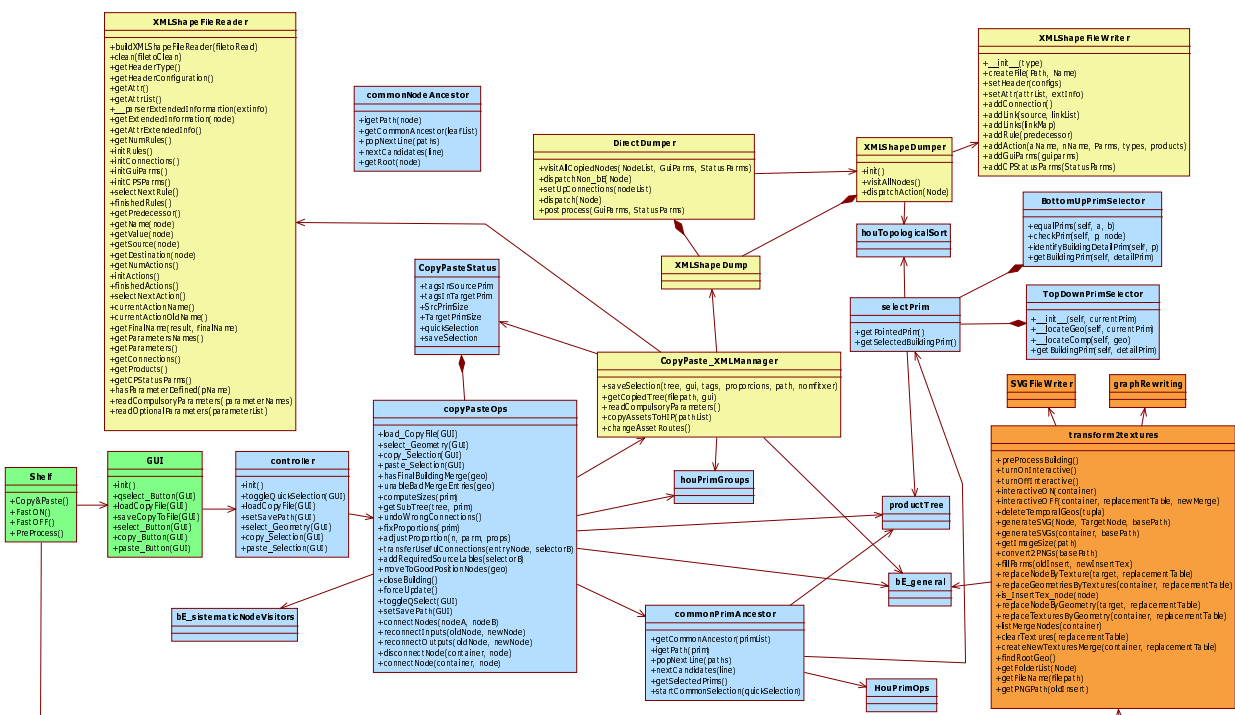


Figura 4: Diagrama de classes complet.

5 Resultats

Seguidament, en la Figura 5, es pot veure un exemple d'una còpia de part d'una façana entre dos edificis amb l'eina implementada de Copy&Paste.

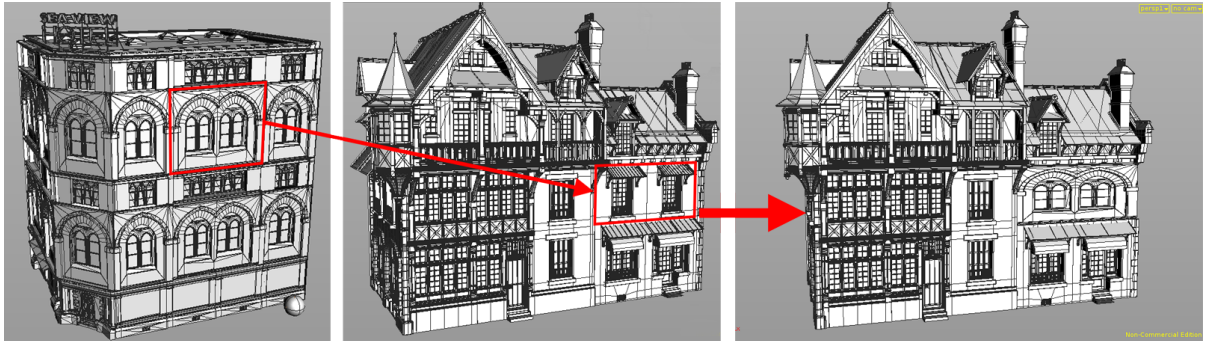


Figura 5: Exemple del resultat de la operació de Copy&Paste amb l'hotel SeaView i la casa Racollette.

A continuació, en la Figura 6 es pot veure una gràfica on es mostra el rendiment del mòdul d'acceleració de la interacció d'usuari, en l'augment iteratiu de l'alçada d'un edifici i el conseqüent increment en la geometria. La gràfica mostra una comparativa sense acceleració (blau) i amb l'acceleració (vermell).

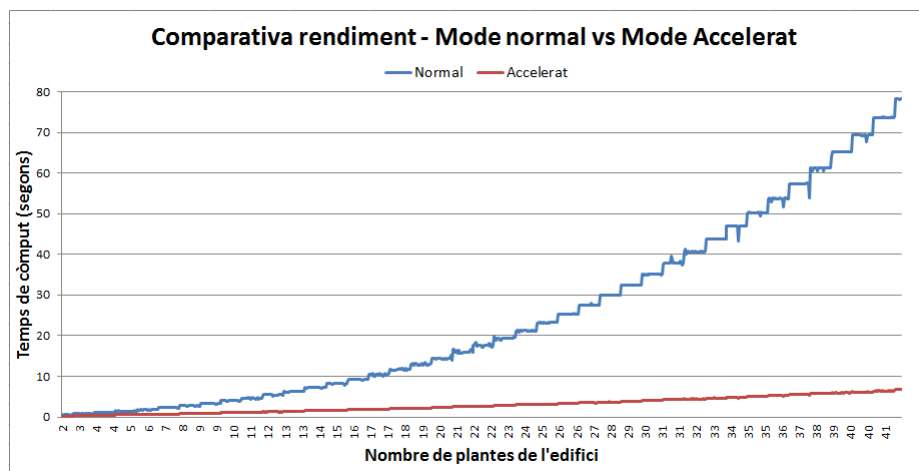


Figura 6: Resultats del test d'acceleració.

6 Conclusions

Durant l'elaboració d'aquest projecte s'han arribat a diverses conclusions de les quals cal destacar:

- S'ha après el llenguatge de programació Python que era desconegut fins al moment de l'elaboració d'aquest projecte.
- L' estudi de la plataforma 3D "Houdini" i de les llibreries per la incorporació de scripts Python. Gràcies a això s'ha pogut elaborar la interfície d'usuari. Per dur a terme aquesta tasca s'ha hagut de conèixer en profunditat les llibreries per a la programació de scripts en python.

- L'estudi del **buildingEngine**, que és el mòdul inclòs al **skylineEngine** encarregat de la definició d'edificis. Ha sigut clau per a la realització d'aquest projecte ja que proporciona tots els nodes involucrats en la selecció, còpia i enganxat de la geometria.
- Com s'explica en el capítol 8, s'ha hagut de construir diverses llibreries relacionades amb el recorregut de nodes i primitives que generen un edifici per tal de poder realitzar la selecció de geometria així com el seu intercanvi per textures. Aquesta part ha resultat molt enriquidora.
- Gràcies a les classes *copyPasteOps* i *commonPrimAncestor*, s'ha aconseguit desenvolupar un sistema de selecció de geometria de la façana d'un edifici fet amb el mòdul de **buildingEngine**.
- La realització de l'algorisme de reconeixement de les primitives involucrades en la generació de la part seleccionada, implementat a la classe *commonPrimAncestor*.
- L'emmagatzemament dels nodes (operacions geomètriques) participants mitjançant un mòdul d'escriptura i lectura de fitxers en format XML amb tota la informació necessària per a poder-los reproduir més endavant. Aquest mòdul s'ha explicat en profunditat en el capítol 9.
- Mitjançant la funció de còpia de la classe *copyPasteOps*, s'ha aconseguit emmagatzemar els nodes seleccionats en el buffer de Houdini amb èxit.
- La reescriptura del graf de destí per incorporar-se les parts copiades. Aquesta part ha sigut una de les funcionalitats clau del projecte ha estat implementada en la classe *copyPasteOps*. Aquesta funcionalitat s'ha basat en l'article de Barroso et al. [1] i s'ha pogut realitzar amb èxit l'algorísmica de la reescriptura del graf destí per incorporar la geometria copiada.
- Gràcies a la implementació de la selecció de geometria s'ha pogut dotar d'una eina que ha permès seleccionar molt més ràpidament la geometria. Ja que no cal que l'usuari conegui l'estructura tècnica que fa possible la visualització de l'edifici.
- Acceleracions en la interacció de l'usuari s'ha realitzat amb èxit, es s'ha vist un exemple en la Figura 6.

Referències

- [1] Santiago Barroso, Gonzalo Besuievsky, and Gustavo Patow. Visual *copy & paste* for procedurally modeled buildings by ruleset rewriting. *Computers & Graphics*, 2013.