



UNIVERSITÉ  
DE MONTPELLIER

---

# GÉNÉRATEUR DE CHÂTEAUX-FORTS

---

Projet destiné à la construction procédurale de châteaux-forts

Réalisé par :

Vincent DE ROUSIERS et Karim EL AOUAJI

Pour :

HMIN 317 Moteur de Jeux

## Table des matières

Introduction.....	4
Présentation du sujet.....	4
Caractéristiques du projet.....	4
Étude architecturale.....	5
Les Mottes (ou Motte Castrale).....	5
Les Châteaux Maçonnés (ou forteresses).....	6
Les résidences nobiliaires.....	8
Méthodes de génération.....	9
Approche naïve.....	9
Approche L-System.....	11
Aspects techniques.....	14
Visualisation.....	14
Camera.....	14
Texture.....	16
Shader.....	16
Panel de contrôle.....	17
Construction.....	18
Placement.....	18
Génération d'une brique de base.....	19
Murs.....	19
Tours.....	20
Joints entre les briques.....	20
Détails.....	21
Conclusion.....	22

## Introduction

Les châteaux-forts, ces immenses constructions datant du Moyen Age, sont des monuments connus de tous et à l'architecture impressionnante. Notre projet souhaitant réaliser l'édition procédurale d'un château-fort, ce rapport va s'attacher à expliquer et illustrer la construction de notre travail et le cheminement du développement de notre programme.

### Présentation du sujet

Notre sujet, « Générateur de châteaux-forts », portait sur deux aspects principaux :

- la génération aléatoire d'un modèle 3D en fonction de divers paramètres ;
- la visualisation de ce modèle 3D.

Plusieurs bonus étaient aussi proposés, tels que texturer le modèle, générer les textures et animer le château.

Notre programme doit donc être capable de créer, à partir d'une liste de paramètres initiaux, un château-fort correspondant aux critères retenus pendant l'étude architecturale.

Ainsi, pour créer ce programme capable de simuler procéduralement l'édition d'un château-fort, il a tout d'abord fallu se renseigner sur les méthodes architecturales ainsi que sur les *patterns* récurrents.

Une fois ce choix architectural établi, la question de la méthode de construction s'est posée. Après quelques essais, nous avons décidé de choisir une approche *L-System*. Celle-ci nous permettra d'obtenir des résultats plus réalistes et correspond à la demande de génération procédurale.

Les outils et les méthodes de constructions choisis, viennent les choix de visualisation et de construction des briques du *L-System*. Nous allons ainsi voir le déroulement et les explications de toutes ces étapes dans ce rapport.

### Caractéristiques du projet

Répondant aux spécifications imposées lors de l'attribution des sujets, notre programme est développé en utilisant Qt (dans sa dernière version, 5.5.1) et la bibliothèque OpenGL3.0. Le développement s'est fait sur l'environnement Linux.

## Étude architecturale

L'étude architecturale de notre projet s'est porté sur trois styles parmi les plus importants des châteaux-forts. Nous commencerons par les prémisses avec les « Mottes », puis nous parlerons du style choisi pour notre projet, et représentatif de l'idée que chacun se fait d'un château-fort, les châteaux maçonnés ou forteresses. Nous finirons cette étude en passant sur la dernière représentation des châteaux-forts, les résidences nobiliaires.

### Les Mottes (ou Motte Castrale)



*Illustration 1: Une motte dans le centre de la France, XI ème siècle*

Une motte castrale, est une fortification de terre, très répandue au Moyen Age. Elle est principalement composée d'une surélévation de terre, le tertre. On trouvait souvent un fossé l'entourant et une palissade de bois formant l'enceinte. Le haut du tertre était occupé par un donjon primitif constituant une tour de guet.

Il s'agissait là des tout premiers châteaux forts, même si il s'agit plus de cabanes fortifiées que de réels constructions en dur. Elles étaient très répandues au début du Moyen Age, de par leur faible coût de construction et de la rapidité à les mettre en place.

Ces constructions n'étant pas très riches architecturalement et manquant cruellement de détails ou de bâtiments emblématiques, il n'était pas très intéressant de se baser sur celles-ci pour notre projet. Nous avons donc continué notre étude et étudié un style plus récent, les châteaux maçonnés.

## Les Châteaux Maçonnés (ou forteresses)



*Illustration 2: Château fort du nord-ouest de la France, XV ème siècle*

Les châteaux maçonnés représentent l'âge d'or de la construction défensive du Moyen Age, ils sont parmi les plus répandus et les plus utilisés. Plus complexes et plus longs à construire que les Mottes, ils sont bien plus résistant aux invasions et permettent aux seigneurs de montrer leur puissance et leurs richesses.

Architecturalement fournis et intéressants à étudier, ils sont composés de nombreux bâtiments intérieurs et leurs enceintes présentent de nombreuses variations. Ils sont presque tous entourés de douves et munis d'un pont-levis. Il s'agit donc du choix le plus intéressant pour nous à utiliser, nous permettant de nous baser sur des constructions précises mais capable de supporter plusieurs variations, ce qui est parfait pour de la génération procédurale.



*Illustration 3: Exemple typique d'un château fort de moyenne importance du XIV ème siècle*

## Les résidences nobiliaires

Les résidences nobiliaires sont la dernière étape de l'évolution des châteaux-forts et le dernier style que nous avons étudié. Elles marquent la fin des demeures fortifiées et une modification de la pensée des seigneurs. Leur fonction n'est plus de défendre mais d'impressionner, d'exhiber au mieux la richesse du propriétaire.

Il s'agit de constructions bien plus raffinées que les précédentes et plus précises, moins brutes. S'agissant plus de demeures et de maisons que de réels châteaux forts, nous avons pourtant décidé de ne pas déborder de notre sujet et de ne pas utiliser ce style. Bien qu'offrant de nombreuses variations (notamment à travers les différentes ailes pouvant composer ces bâtiments) et pouvant produire des résultats plus jolis que ceux produit avec le style des forteresses, ces constructions ne ressemblent plus assez à l'idée que l'on se fait d'un château-fort.



*Illustration 4: Le Château de Versailles, XVII ème siècle*

Cette étude architecturale, nous a permis de voir de nombreux types de constructions et l'évolution des châteaux-forts à travers l'histoire. Nous avons décidé de nous pencher uniquement sur les « vrais » châteaux-forts, les châteaux maçonnés, bruts et immenses. Ce sont eux qui nous permettront de réaliser les modèles 3D les plus intéressants.

## Méthodes de génération

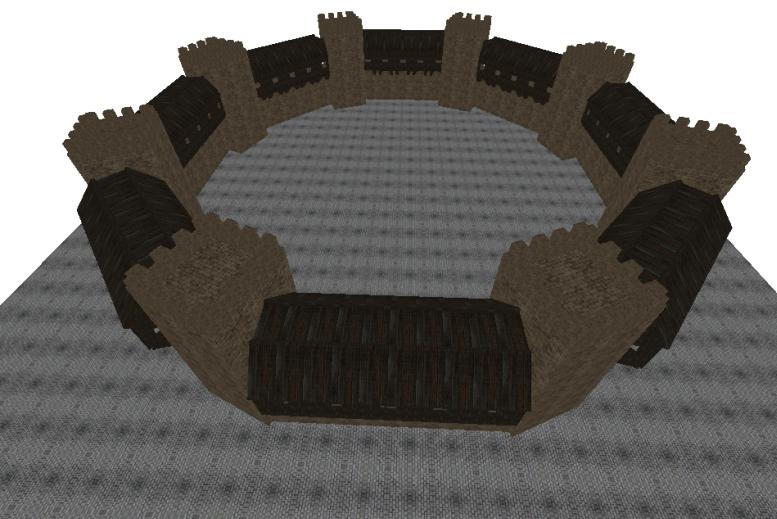
Comme nous l'avons appris durant notre étude bibliographique (sur la génération procédurale de villes), avant de générer les détails et les bâtiments, il faut commencer par générer le « plan » sur lequel poser les briques de constructions. Nous avons utilisé deux approches différentes.

### Approche naïve

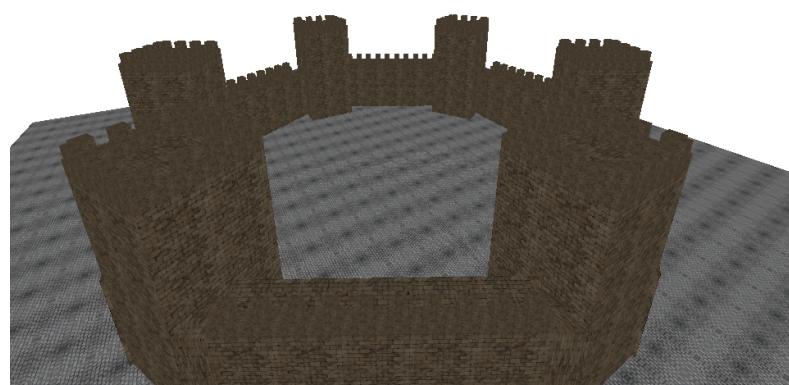
La première méthode utilisée, était une approche assez simple. En partant d'un point central, on construisait en cercle l'enceinte du château. Cette enceinte est constituée d'une suite de murs et de tours, elle est fermée. Nous obtenions ainsi une enceinte homogène, dont nous pouvions aisément augmenter le nombres de côtés. Malheureusement, dès que nous voulions permettre aux différents murs et tours d'adopter des tailles différentes, cette méthode se révéla incapable de répondre à nos besoins.

Très vite, cette méthode nous a donc paru très limitée et gênante pour notre génération. Elle nécessitait beaucoup de calcul, de prendre en compte de nombreux cas particuliers et ne fournissait que des résultats assez décevants (sauf si l'on désire avoir une enceinte homogène).

Nous l'avons donc mise de côté au profit d'une seconde approche. Néanmoins, elle pourrait de nouveau être utilisée pour former des enceintes aux motifs répétés, typiquement pour les forts en V (constructions bien plus récentes que celles que nous étudions).



*Illustration 6: Génération d'une enceinte de 8 côtés par approche naïve*



*Illustration 5: Génération d'une enceinte de 6 côtés par approche naïve, détails différents de la précédente*

## Approche L-System

Plus complexe à mettre en œuvre mais fournissant des résultats plus réalistes, il s'agit de la méthode que nous avons finalement choisi pour notre projet. Nous l'avions déjà étudié dans notre rapport bibliographique et il s'agit d'une méthode couramment utilisée dans le monde de la génération procédurale.

Nous partons toujours d'un point central, qui va servir de centre au château et accueillir les bâtiments intérieurs, et nous construisons l'enceinte autour. La construction n'est cette fois plus uniquement guidée par le centre du cercle et son rayon mais par la taille des briques qu'on lui ajoute (les briques étant les murs et les tours).

L'algorithme prend comme entrées principales les tailles minimales et maximales des tours et murs ainsi que du nombre de murs. Il commence alors par décider du nombre de murs dont l'enceinte sera composée puis place une tour à distance du centre. Une fois le nombre de murs posé et la tour placée, l'algorithme va rajouter un mur puis une tour puis un mur etc., jusqu'à atteindre le nombre de murs demandés.

Chaque pose de mur ou de tour va devoir respecter plusieurs paramètres. Les murs et tours doivent avoir des tailles comprises entre les bornes données en entrées et ainsi placés à la bonne distance de la brique précédente (afin de ne pas voir de collisions ou de chevauchements). Il faut aussi faire subir aux briques une rotation, en fonction de l'angle entre deux murs.

Cet angle est calculé en fonction du nombre de murs, si l'enceinte doit comporter 4 murs, tous les angles seront compris entre deux angles bornes. Ces angles bornes sont calculés en fonction de l'angle de base (pour 4 murs, 360 divisé par 4, 90) auquel on peut ajouter ou soustraire un certain pourcentage de cet angle de base, le degré de liberté. Si le degré de liberté est de 10 (10%) et le nombre de côtés 4, tous les angles seront compris entre 81 et 99.

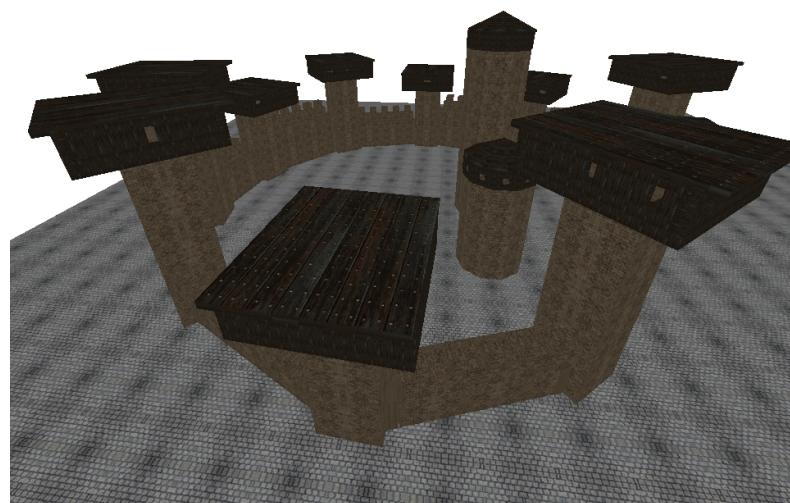
Cette approche permet donc d'obtenir une suite réaliste de murs et de tours de tailles variées et ne se chevauchant pas. Il reste encore à relier et combler les petits écarts entre ces briques et leur ajouter des détails, ce que nous expliquerons dans la suite de ce rapport.



*Illustration 7: Génération par L-System*



*Illustration 8: Génération par L-System, différente de la précédente*



*Illustration 9: Génération par L-System, encore différente*

## Aspects techniques

Cette partie portera sur l'explication du développement de nos fonctionnalités principales.

### Visualisation

#### Camera

Afin de permettre une bonne visualisation de notre génération, nous avons logiquement choisi de permettre à l'utilisateur de contrôler une caméra dite *free-fly*. Celle-ci l'autorise à se promener librement dans l'ensemble du modèle et de l'observer depuis n'importe quel point de vue. Les déplacements de cette caméra se font à l'aide de la souris et des touches Z,Q,S et D.

#### Texture

Dans le but d'obtenir un modèle lisible et d'identifier facilement les différents éléments, il est important de fournir plusieurs textures. Notre programme est conçu de telle sorte qu'il est très aisément d'ajouter une texture et de l'utiliser.

Chaque texture est lié à un indice et chargée lors de l'initialisation du programme, pour qu'une fonction l'utilise, il suffit qu'elle fasse appel à l'indice de la texture. On peut donc très facilement modifier la texture utilisée pour n'importe quelle brique du modèle.

#### Shader

Les châteaux-forts étant constitués d'une multitude de briques (dans le sens de la maçonnerie cette fois-ci), il est impératif de leur donner un effet de profondeur. Nous avons pour cela utilisé la technique du *Normal Mapping* via nos shaders. Si la texture utilisée possède une *normal map* (chargée comme une texture normal mais relié à sa texture de base), le shader va s'en servir pour créer l'effet de profondeur, dans le cas contraire il se contente d'appliquer le modèle de Phong pour l'éclairage.



Illustration 10: Texture du sol, sans  
normal mapping



Illustration 11: Texture des murs,  
avec normal mapping



Illustration 12: Texture bois, sans normal  
mapping

## Panel de contrôle

Dans le but de permettre à l'utilisateur de pouvoir modifier à la volée les paramètres de la simulation et d'en observer les résultats en temps réel, nous avons fournis, en complément de notre générateur, un panel de contrôle.

Ce panel permet de modifier les points essentiels de la simulation, tels que :

- le nombre de murs constituant l'enceinte ;
- les différentes dimensions des murs et des tours ;
- les détails que l'on affecte aux murs et aux tours ;
- le rayon moyen de l'enceinte ;
- la graine d'initialisation du random.

The screenshot shows a user interface for generating simulation parameters. At the top, there is a 'Seed' field containing the value '147' with a small orange slider below it. Below this, there is a section titled 'Nombre de murs de l'enceinte' (Number of walls in the enclosure) with 'Min' and 'Max' fields both set to '8'. Further down, there is a 'Radius' field set to '10' and a 'Détails' (Details) field set to '3'. A section titled 'Paramètres des murs' (Wall parameters) follows, with 'X : Min' and 'Max' fields both set to '2', 'Y : Min' and 'Max' fields both set to '3', and 'Z : Min' and 'Max' fields both set to '1'. There is also a 'Details' field for walls set to '2'. At the bottom, there is another section titled 'Paramètres des Tours' (Tower parameters), which has identical settings to the wall parameters: 'X : Min' and 'Max' fields both set to '2', 'Y : Min' and 'Max' fields both set to '3', and 'Z : Min' and 'Max' fields both set to '1'. There is also a 'Details' field for towers set to '3'.

Illustration 13: Panel de contrôle  
du générateur

## Construction

Cette sous partie va détailler les techniques utilisées pour construire les briques principales de notre modèle. Nous parlerons notamment de la constructions des murs, des tours, des joints entre les briques et de l'ajout des détails. Mais passons tout d'abord à la méthode de placement de nos briques et la génération d'une brique de base

### Placement

Tout ce qui est construit dans notre programme possède trois éléments principaux :

- l'état précis de l'endroit où l'objet est construit, son repère, il s'agit d'une matrice 4x4 comportant les déplacements, rotations et échelles actuels du monde ;
- les déplacements par rapport à l'endroit précédent, un vecteur 3 ;
- les rotations par rapport à l'endroit précédent, un vecteur 3 ;
- les échelles par rapport à l'endroit précédent, un vecteur 3 ;

Ainsi, si l'on désire construire une tour orienté à 45 degré sur l'axe X, se déplaçant de 5 sur l'axe Z et dont l'échelle est de 2 sur les trois axes, tout ce que cette tour va produire par la suite (ses détails par exemple) se trouvera dans le même repère, un objet transmettant son repère à ce qu'il va construire.

Cette méthode est très pratique pour le programmeur, car il est très simple ainsi de rajouter des éléments à un objet existant en se basant sur le repère propre de l'objet. Il faut juste s'assurer que l'objet transmette bien son repère à ses fils.

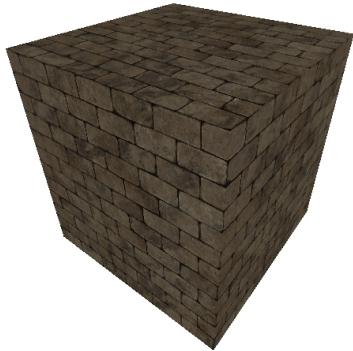
```
QMatrix4x4 matGlobale, QVector3D translate, QVector3D rotate, QVector3D scale,
```

*Illustration 14: Les paramètres permettant de placer les briques créées*

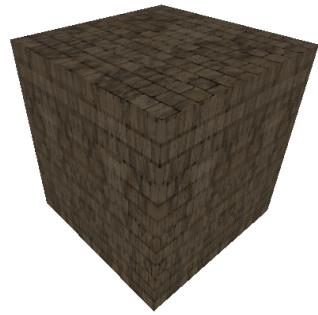
## Génération d'une brique de base

Pour créer un mur, il faut créer un polygone. Ce polygone est composé de points. Nous avons donc créé une méthode à laquelle on fournit un nombres de points et leurs coordonnées, un nombre de faces et la répétition de la texture utilisée sur cette face (afin de ne pas avoir de texture mal positionnée). Cette méthode va ensuite construire un polygone à partir des points donnés.

Cette méthode permet de construire très aisément des cubes, pavés ou autres et donc les éléments de notre modèle, comme les murs et les tours ou bien les détails.



*Illustration 15: Exemple d'un cube bien texturé*



*Illustration 16: Exemple d'un cube mal texturé*

## Murs

Les murs sont construit en respectant la méthode précédente, on place le centre du mur au centre du repère parent au mur (si il en possède un). A partir de ce centre, on va créer une brique dont la taille est définie sur les trois axes par les caractéristiques propres à ce mur. Les points composant le mur sont calculés en fonction du centre et des dimensions.

Par exemple, si on est dans un repère placé en 0,4,0 et que le mur possède comme dimensions 4,2,1 ; le mur s'étendra dans le repère global de -2 à 2 dans l'axe X, de 3 à 5 dans l'axe Y et de -0,5 à 0,5 dans l'axe Z.

## Tours

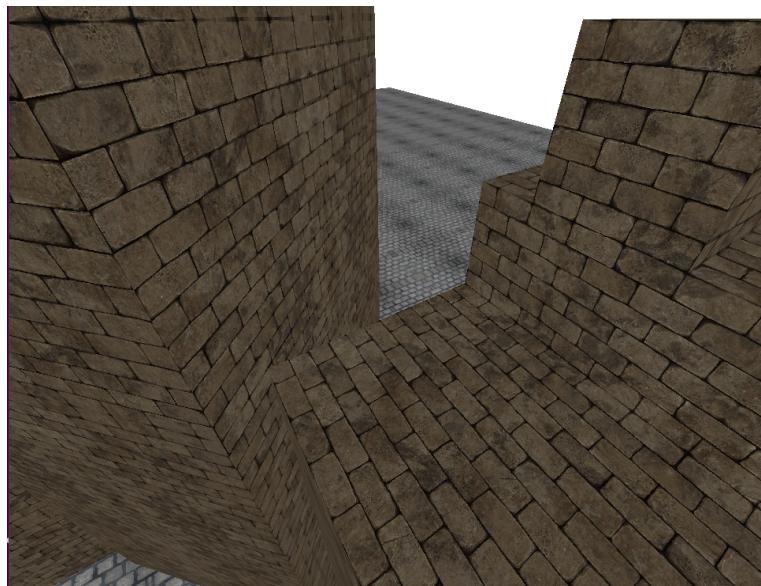
Intuitivement, une tour n'est qu'un mur qui est plus haut que long, ainsi on utilise la même méthode pour construire un mur qu'une tour, la différenciation des deux se faisant par la suite via les détails (que nous verrons bientôt).

Une seule exception subsiste dans notre programme, les tours rondes. Celles-ci ne sont pas construites comme des pavés mais comme des cylindres, assez logiquement.

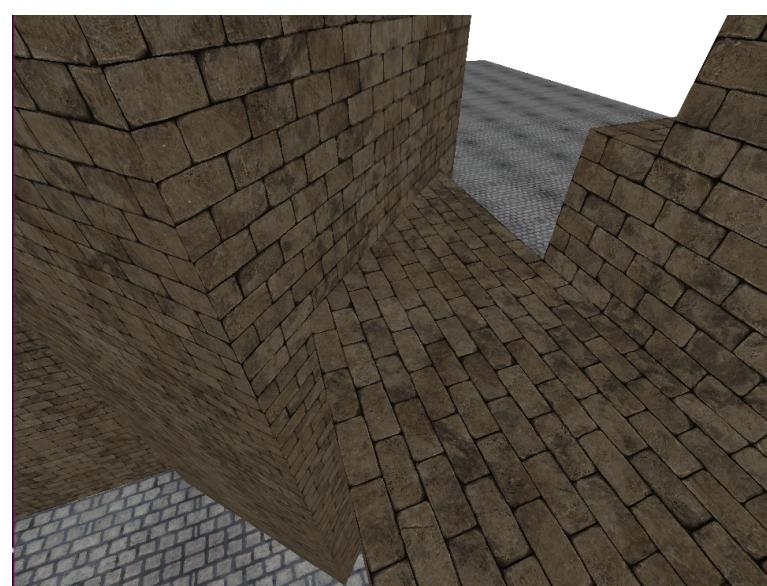
Elles sont donc principalement caractérisées par un rayon, une hauteur et le nombre de points composant le diamètre. Si il y a 4 points, il s'agit d'un pavé par exemple.

## Joints entre les briques

Quand le *L-System* construit l'enceinte, la jonction entre un mur et une tour laisse fréquemment un blanc. Pour combler ce blanc, il faut construire un joint entre les deux briques. Ce joint est construit en générant un polygone de 6 points placé entre une tour et un mur.



*Illustration 17: Exemple d'un creux entre un mur et une tour dans l'enceinte*



*Illustration 18: Ajout d'un joint pour combler le creux*

## Détails

Les détails (les ajouts sur les briques des murs ou des tours, tels les créneaux) sont un aspect très important de la génération procédurale. Ils permettent de varier les modèles créés et de rendre chaque génération unique. En effet, on peut choisir de fixer les détails ou de laisser un certain degré de liberté au programme, créant une architecture plus ou moins homogène.

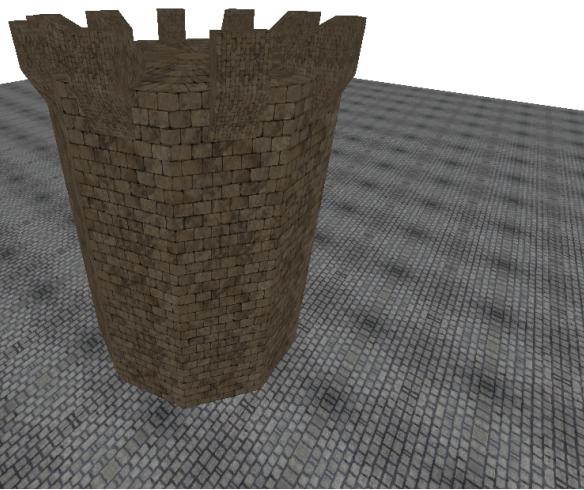


Illustration 19: Tour ronde à créneaux vers l'extérieur



Illustration 20: Tour ronde avec hourd complet

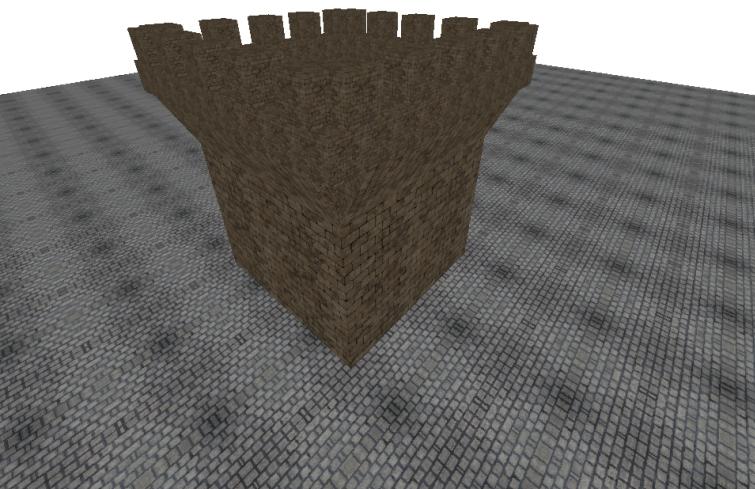


Illustration 21: Tour carrée avec créneaux vers l'extérieur

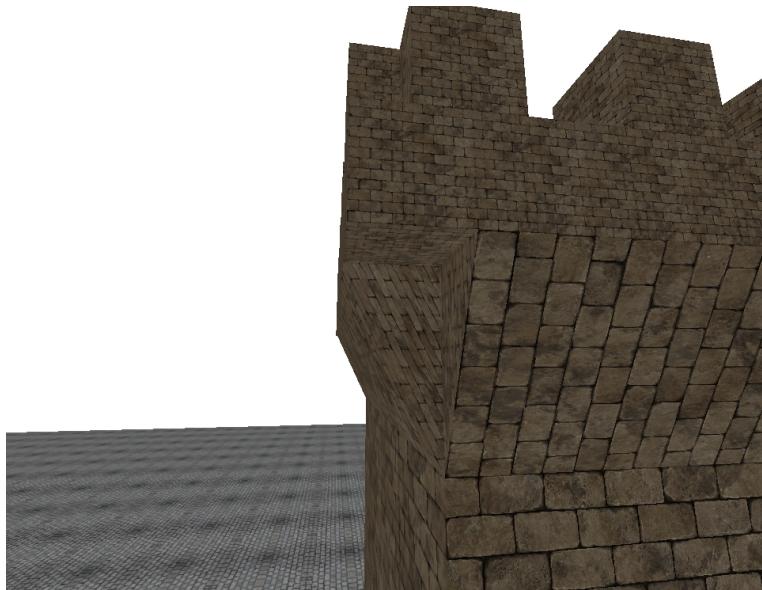


Illustration 22: Détails de la tour précédente

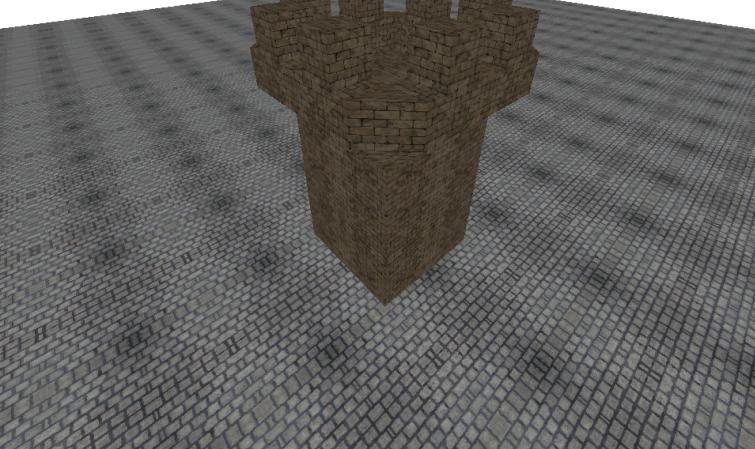


Illustration 24: Tour carrée avec détails différents

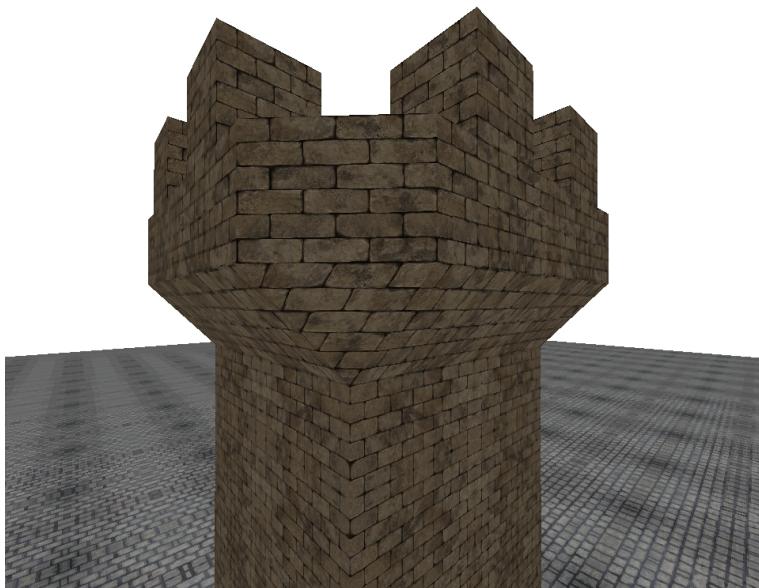


Illustration 23: Détails de la tour précédente

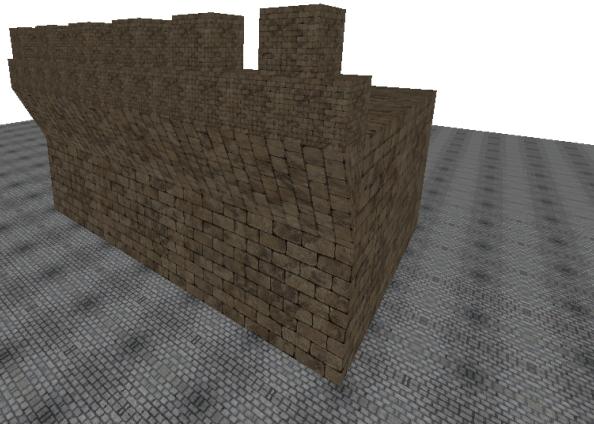


Illustration 25: Mur avec créneaux vers l'extérieur

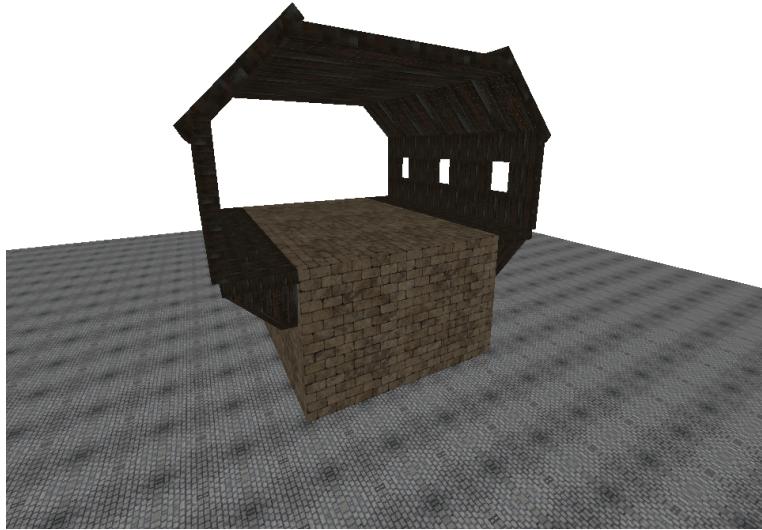


Illustration 26: Mur avec hourd complet

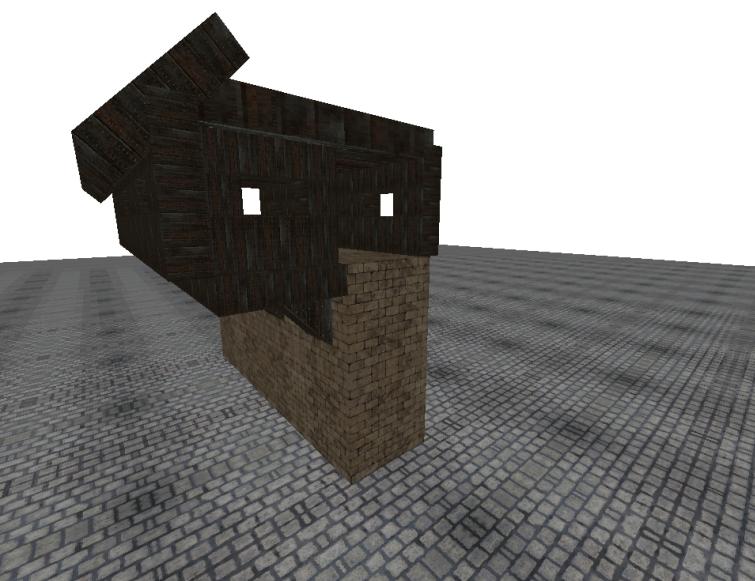


Illustration 27: Mur avec hourd vers l'extérieur

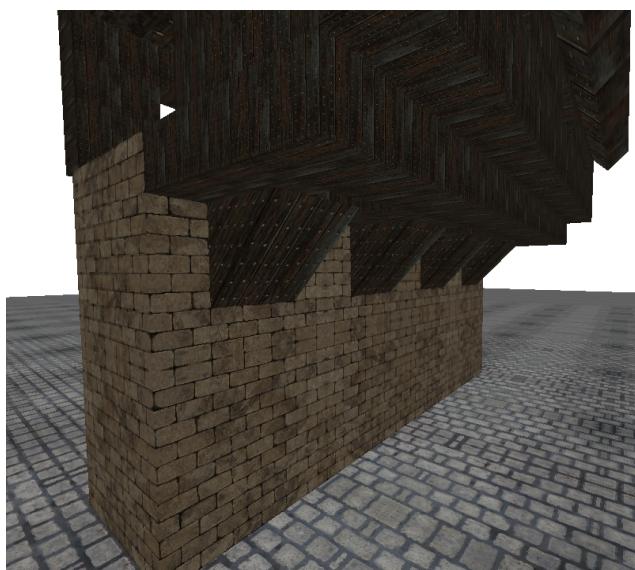


Illustration 28: Détails du mur précédent

## Conclusion

Notre projet permet la génération procédurale d'une enceinte de château-fort ainsi que d'un donjon. Il reste néanmoins de nombreux ajouts à effectuer pour le rendre pleinement complet. Il serait nécessaire de finaliser la construction du pont levis et son intégration à l'enceinte. Ajouter des bâtiments intérieurs serait aussi un énorme plus mais nécessiterait beaucoup de travail de modélisation.

Tout ces ajouts sont possibles et faciles à réaliser grâce aux outils développés mais nécessiteraient tout de même un certain temps de travail. Tout comme créer de nouveaux détails pour les tours et les murs, cela se ferait sans impacter le L-System déjà implémenté et l'enrichirait en apportant de nouvelles variations.

En l'état, notre programme fournit à l'utilisateur tous les outils pour créer des briques, que l'on peut modeler comme on le souhaite, et les assembler pour un château-fort basique. Notre méthode de génération est paramétrable en temps réel et permet à l'utilisateur de modifier sa construction à la volée comme il le souhaite.