

# Éviter les problèmes de sons ( v1.0 ) AT AN

## **Problème:**

On tire sur un mur opposé et on n'entend pas de son d'impact.

Le problème est dû au design du niveau et on peut l'éviter en se rappelant de quelques points pendant qu'on est en train de construire son niveau.

## **Quelques bases préalables: (désolé, c'est nécessaire)**

Pendant qu'il crée le BOA Vis, le D3Edit calcule les distances de sons pour le niveau et en fait une liste. Cette liste est sauvegardée avec le niveau et Descent 3 l'utilise pendant le jeu pour créer les sons.

Quand il calcule les distances, le D3Edit vérifie si la distance est plus grande que 400 unités.

Quand la distance excède cette limite, l'entrée de la distance est déclarée invalide.

Si Descent 3 trouve une entrée de distance invalide, il ne va pas générer de son d'impact.

## **Comment cette liste de distances est-elle calculée?**

Le D3Edit vérifie pour chaque pièce quelle est la pièce la plus éloignée "dans laquelle on peut voir".

Pendant qu'il le vérifie, le D3Edit additionne les distances entre la pièce de départ et la pièce la plus éloignée.

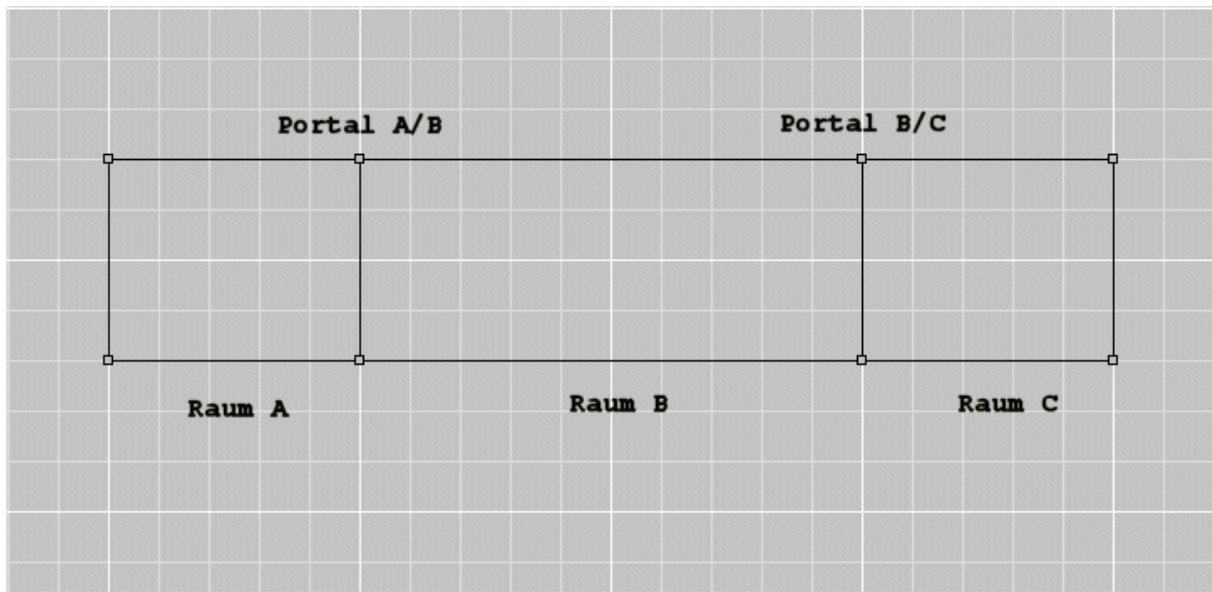
Puis cette distance est placée dans la liste si elle est valide.

## **Comment marche le "voir d'une pièce à l'autre"?**

Étudions un niveau exemple. On a ici 3 Pièces alignées (A, B, C) qui sont connectées par des portails.

(Room A -> B; Room C -> B; Room B -> A and Room B -> C)

Ainsi, la Room A a *un* Portail, la Room B *deux* Portails et la Room C *un* Portail.



Le D3Edit calcule d'abord pour chaque pièce la ligne de mire max.

Et il insère pour chaque Pièce (c'est la pièce de départ) la Pièce d'arrivée dans le tableau.

Pour notre niveau exemple, on obtient le tableau suivant:

Pièce de départ	Pièce d'arrivée
A	C
B	A
B	C
C	A

## Comment cette "distance de Pièce à Pièce" est-elle alors calculée?

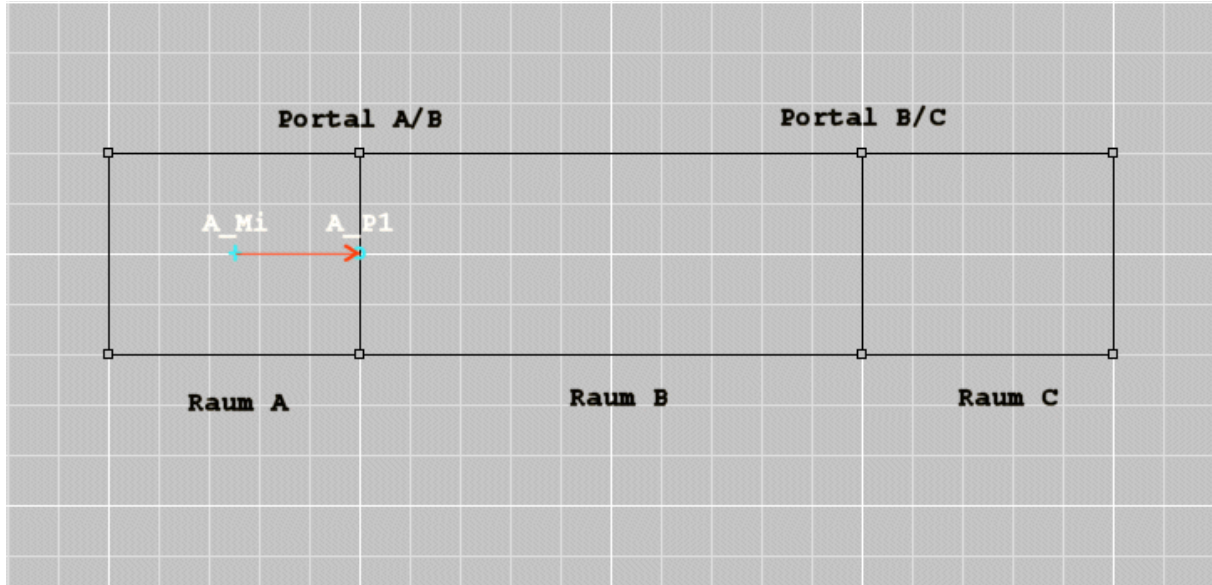
Voyons ce que le D3Edit fait avec la Room A comme pièce de départ.

Selon le tableau généré ci-dessus, la distance est calculée du centre de la Room A au centre de la Room C.

La première étape consiste à calculer la distance entre le centre de la Room A et le premier portail de la Room A.

La distance AC est paramétrée à 0.

Puis une ligne droite est tracée depuis le centre de la Room A jusqu'au centre du portail A/B.



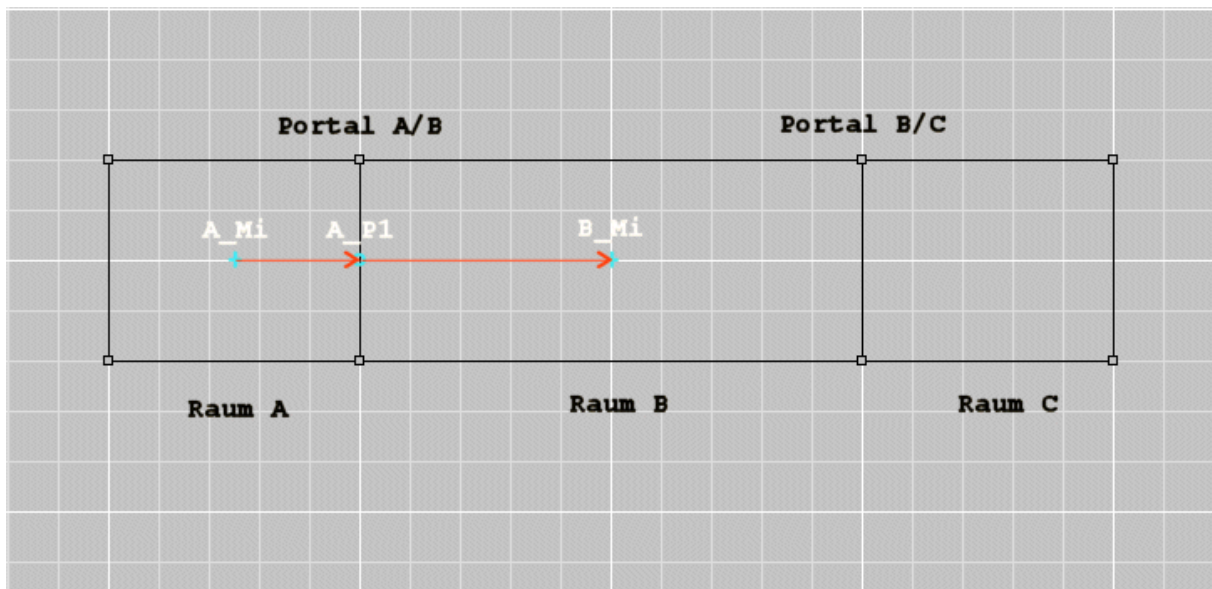
Si cette ligne ne peut pas toucher la face du Portail la distance est invalide.

Distance AC = invalide.

En cas de réussite, la distance est enregistrée.

Distance AC = distance AC + (distance A\_Mi -> A\_P1)

L'étape suivante consiste à tracer une ligne droite depuis le centre du Portail A/B jusqu'au centre de la Room B.



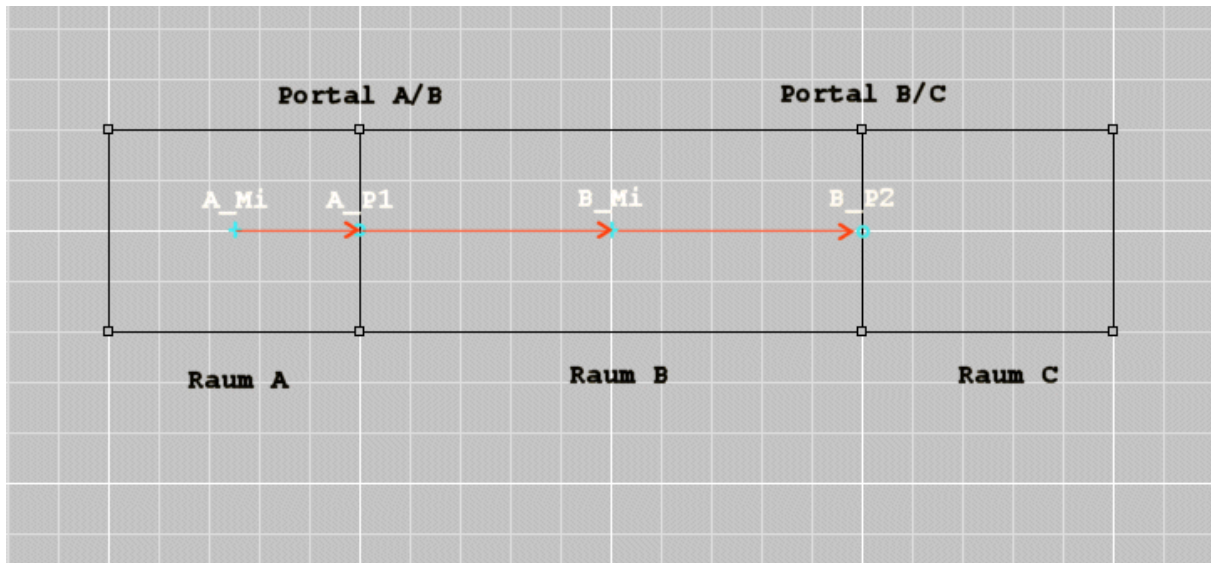
Si cette ligne ne peut pas atteindre le centre de la room B, la distance est invalide.

Distance AC = invalide.

En cas de succès, la distance obtenue est ajoutée à la distance déjà sauvegardée.

Distance AC = distance AC + (distance A\_P1 -> B\_Mi)

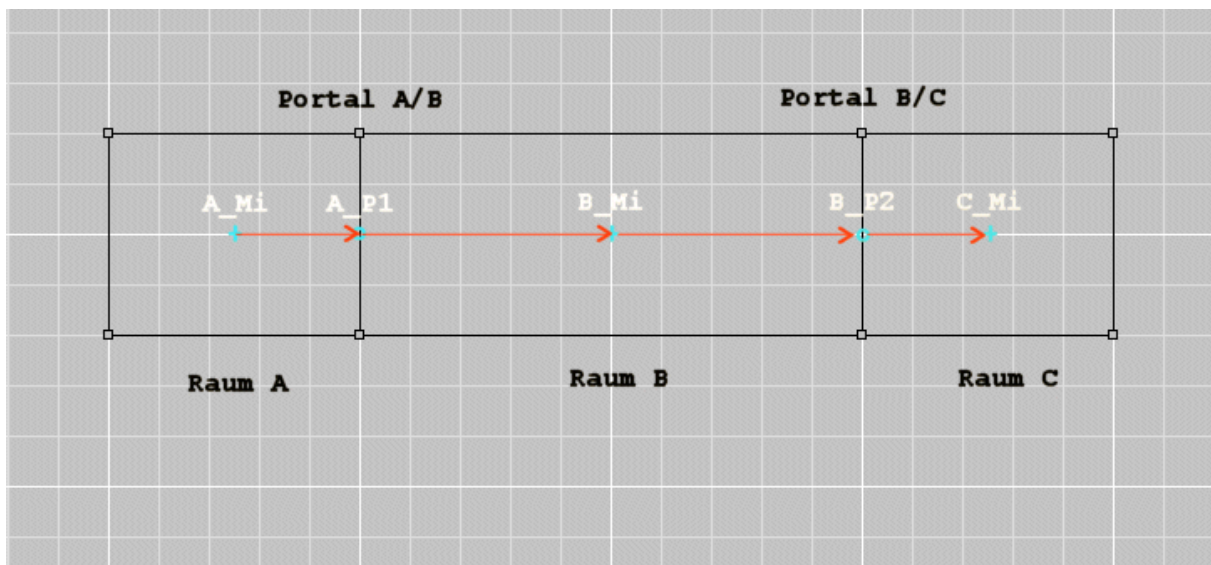
L'étape suivante consiste à tracer une ligne droite depuis le centre de la Room B jusqu'au centre du portail B/C.



Si cette ligne ne peut pas atteindre le centre du portail B/C, la distance est invalide.  
Distance AC = invalide.

En cas de succès, la distance obtenue est ajoutée à la distance déjà sauvegardée.  
Distance AC = distance AC + (distance B\_Mi -> B\_P2)

L'étape suivante consiste à tracer une ligne droite depuis le centre du Portail B/C jusqu'au centre de la Room C.



Si cette ligne ne peut pas atteindre le centre de la Room C, la distance est invalide.  
Distance AC = invalide.

En cas de succès, la distance obtenue est ajoutée à la distance déjà sauvegardée.  
Distance AC = distance AC + (distance B\_P2 -> C\_Mi)

Maintenant, le D3Edit vérifie si la distance AC > 400.

Si c'est le cas:

Distance AC = invalide.

Sinon, la distance obtenue est enregistrée dans la liste en tant que A/C.

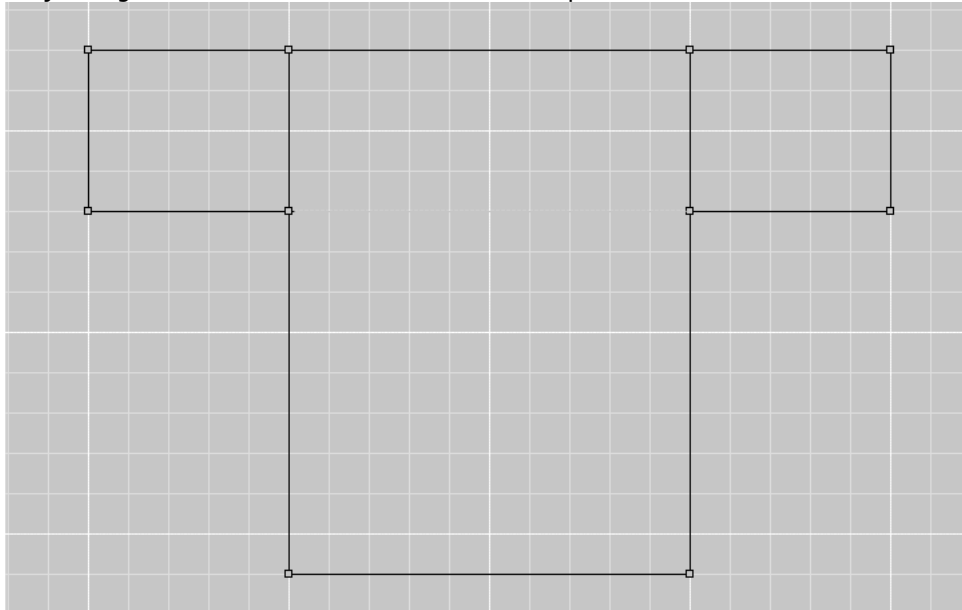
Il n'y avait qu'un seul portail à vérifier pour la Room A, alors le calcul des distances pour la Room A est terminé.

Le D3Edit passe à la Room B et calcule, de la même façon qu'il l'a fait pour la Room A, les distances B/A et B/C. Une fois que le calcul de la distance pour la Room C est fini, toutes les distances max pour les sons sont prêtes. Ces valeurs sont sauvegardées dans le niveau quand on l'enregistre. Si Descent 3 charge le niveau, elles sont utilisées.

### **OK, mais quel est le problème alors?**

Dans l'exemple ci-dessus, on pouvait entendre un impact quand on tirait de la Room A dans la Room C si la distance calculée était  $< 400$ . Sinon, non. (trop loin pour entendre quelque chose).

Mais observons un autre design de niveau maintenant:  
J'ai juste agrandi la Room B comme le montre l'exemple ci-dessous.

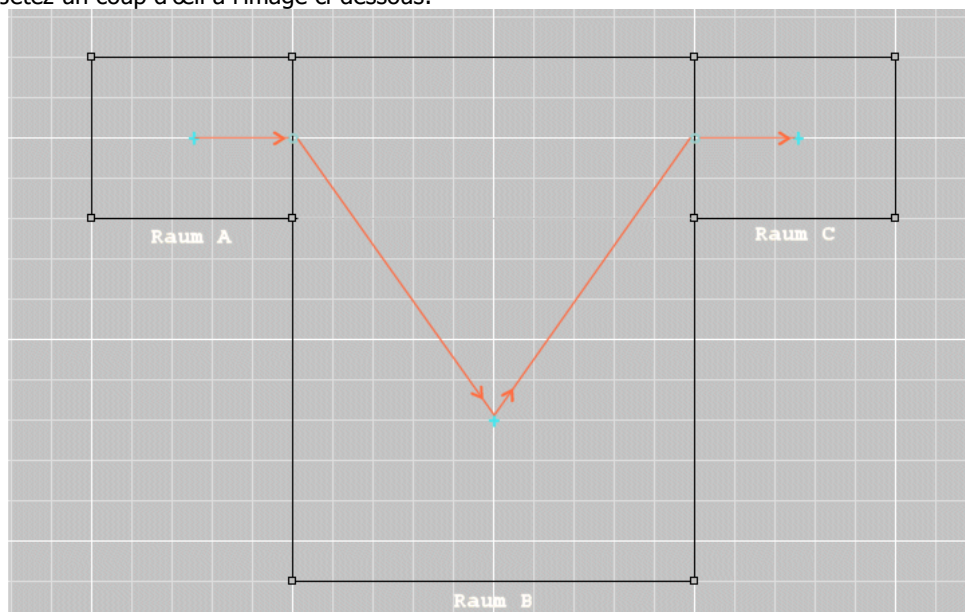


Comme avant, on tire depuis A dans C mais.. on n'entend pas le son de l'impact depuis la Room C maintenant!  
Un autre tir, directement sur le sol devant le Portail B/C, et on entend l'impact à nouveau.  
Passons dans la Room B et tirons dans la Room C. On entend l'impact.  
Tirons maintenant depuis la Room C dans la Room A, pas de son d'impact audible à nouveau!  
Un second tir direct devant le Portail B/A est audible, de même si on tire depuis la Room B dans la Room A.

Qu'est-ce qui s'est passé? La distance entre la Room A et la Room C n'a pas changé. Alors, quoi?

C'est le moment de se rappeler qu'on peut mesurer les distances dans Descent 3 avec le Massdriver.  
Ainsi on peut voir qu'entre la Room A et la Room C, il y a une distance de 250 unités.  
C'est tout à fait OK, c'est en dessous de la limite magique des 400 unités, mais pourquoi n'entend-on pas l'impact?

La réponse est très simple en tenant compte de la façon dont les distances max pour les sons sont calculées.  
Jetez un coup d'œil à l'image ci-dessous:

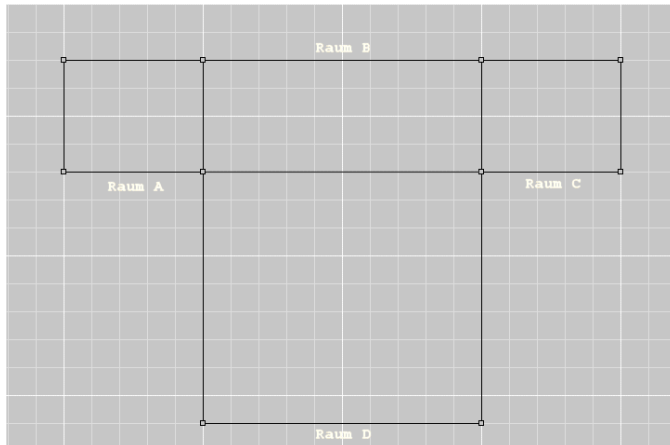


J'ai déplacé le centre de la Room B un petit peu plus bas pour qu'on voie mieux le problème.  
 Vous vous rappelez? Les distances sont calculées depuis le centre des pièces jusqu'aux Portails suivants.  
 En fait c'est exactement là qu'est le problème maintenant. Bien que la distance en ligne droite entre la Room A et la Room C dans cet exemple soit juste de 250 unités, le D3Edit calcule plus du double de la distance, comme on peut le voir facilement sur l'image. C'est pour ça que la distance fait plus de 400 unités. Distance AC = invalide.  
 C'est pourquoi il n'y a pas de son d'impact.

### **Est-ce qu'il y a une solution?**

Oui, il y en a une. Il faut changer le design du niveau.  
 La façon la plus simple est de couper la Room B et de produire une nouvelle pièce Room D attachée à la Room B.

Cette image montre le niveau après le changement:

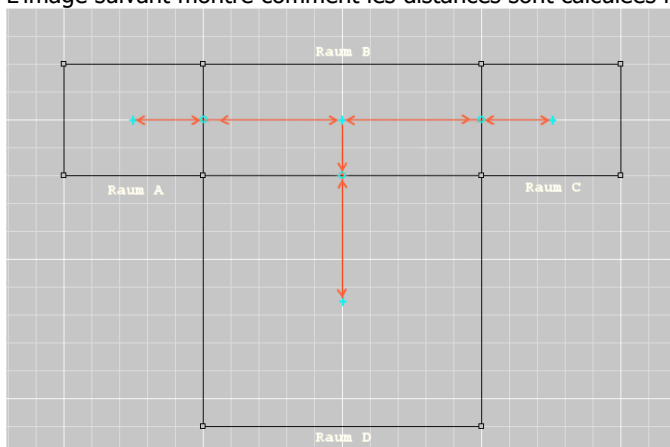


Le D3Edit génère un nouveau tableau de lignes de mire pour ce niveau:

Pièce de départ	Pièce d'arrivée
A	C
A	D
B	A
B	C
B	D
C	A
C	D

Dans le niveau, on peut maintenant entendre les impacts de tirs de A à C et de C à A.  
 Dans le jeu, on ne verrait aucune différence de design entre les niveaux, le nouveau ressemble à l'ancien.

L'image suivant montre comment les distances sont calculées maintenant.



Ça marche parce que la ligne entre la Room A et la Room C fait maintenant moins de 400 unités.

### **Un autre problème...**

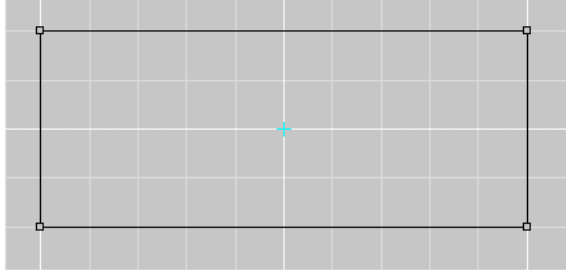
Il y a une fonction du D3Edit qu'on n'utilise pas souvent et qui pourtant nous aide à éviter les problèmes de sons. C'est la fonction -Show Room Center-.

Si elle est activée, elle nous indique par une petite croix la place où Descent 3 voit le centre de la pièce.

Elle se trouve dans chaque vue Ortho et elle indique le centre de masse.

Pourquoi je dis 'centre de masse' et pas Centre de Pièce, ou milieu?

Jetez un coup d'œil aux images suivantes:



Dans cette pièce, le 'centre de masse' est au centre/milieu comme on s'y attendait.

Scindons maintenant une des faces de la pièce:



Bien que la pièce ait gardé la même forme qu'avant, le 'centre de masse' s'est déplacé vers les nouveaux sommets.

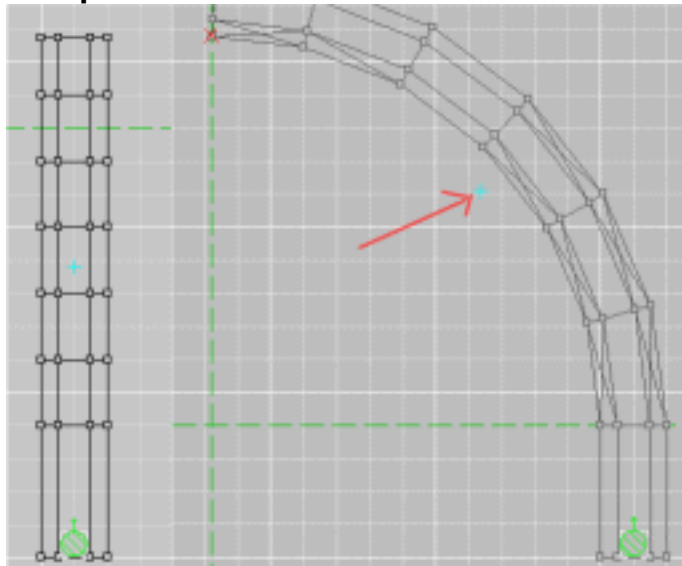
Calculer le centre de la Pièce, c'est plutôt calculer le centre de masse.

Une face avec un grand nombre de sommets va attirer le 'centre de masse' vers elle.

Cette particularité peut produire des défauts pour le son, elle peut aussi nous aider à les éliminer.



### Exemple:

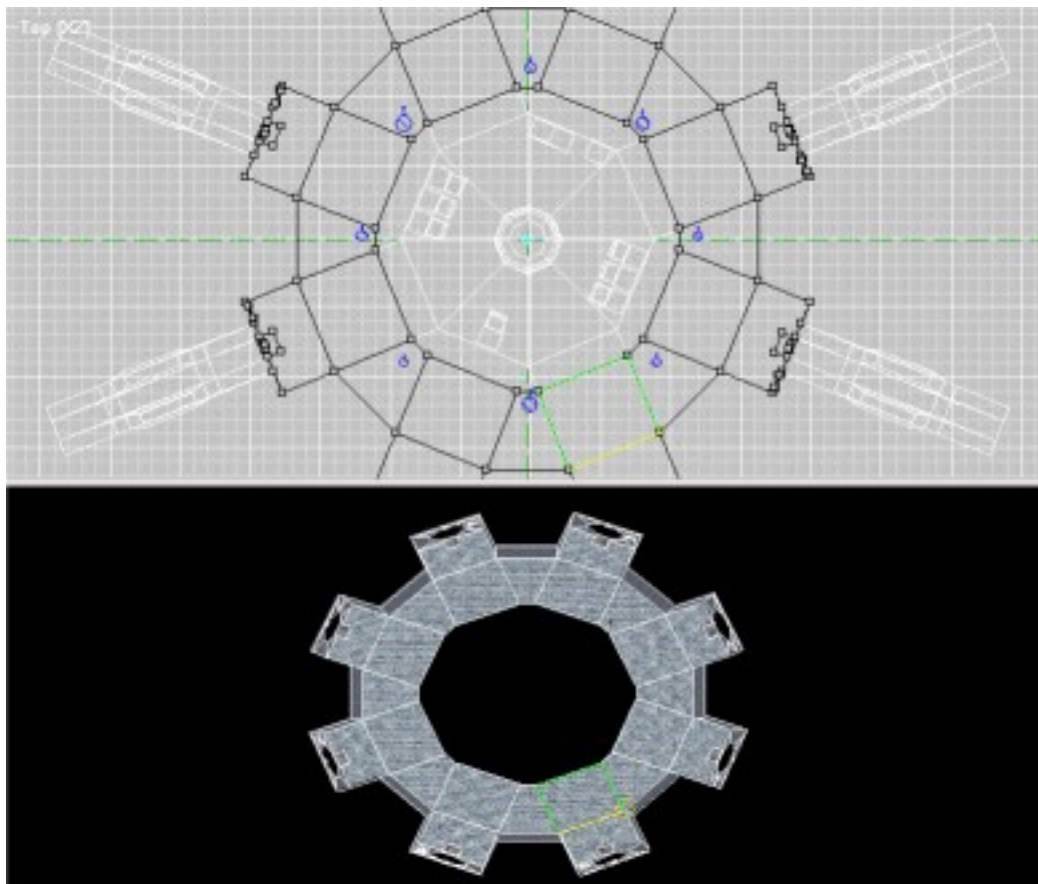


Quand on a courbé ce tunnel, le 'centre de masse' s'est positionné à l'extérieur de la pièce. (flèche rouge)  
C'est fatal pour le calcul des sons. On n'a plus de ligne droite depuis les Portails jusqu'au centre de la Pièce. Une solution serait de découper le tunnel courbe en plusieurs morceaux. Mais, quand la distance entre le 'centre de masse' et l'intérieur du tunnel n'est pas trop grande, on peut se contenter d'insérer des sommets du côté opposé du tunnel pour déplacer ce 'centre de masse'.

On peut aussi insérer un moniteur et utiliser la fonction- Merge Object into Room-.

Les faces et sommets ainsi obtenus peuvent remettre le 'centre de masse' à l'intérieur du tunnel, et le son marchera à nouveau.

Un bel exemple de son qui ne marche pas:



Le 'centre de masse' est parfaitement au milieu, mais il n'est pas à l'intérieur de la bonne pièce ;)

### **Portails:**

Les versions récentes du D3Edit tiennent compte du fait que les textures de portail doivent être perméables au son pour éviter les problèmes de sons dans les niveaux, alors la texture Palmleaf1 est attribuée automatiquement aux faces de Portails.

C'est la seule texture du stock de textures de Descent 3 qui soit perméable pour les lignes de mire et qui autorise un calcul correct du son. Si on veut arrêter les sons à un portail, on choisit une texture différente pour la(les) face(s) de ce Portail.

Rappelez-vous que ces considérations au sujet des textures de Portails sont valables aussi pour les BNodes.