

## Mise en place des systèmes de particules

### Les proies

Choisissez le cube *Proies*. Dans le menu *Object [F7]*, dans les *Particles buttons*, cliquez sur le bouton *Add New* du panneau *Particle System*. Changez quelques-unes des valeurs :

- *Amount* : 100 afin de définir un essaim de 100 particules proies ;
- *Life* : 1500 afin de donner à chaque particule une durée de vie suffisante pour que la simulation proies-prédateurs soit fonctionnelle.

Décidons maintenant du nombre de particules proies qui apparaîtront : dans le panneau *Visualization*, assurez-vous que chaque particule soit représentée comme un *Point* et donnez-lui une taille *Draw Size* de 3 (il s'agit seulement de la taille de la particule dans la vue 3D). Nous voulons que les proies soient petites, aussi, dans le panneau *Extras*, spécifions *Size* égal à 0.3 (il s'agit de la taille de la particule prise en compte dans la simulation ; ce paramètre aide les proies à savoir à quelle distance elles sont les unes des autres, et donc à rester le plus groupées possibles). Il est également temps d'activer l'option *Die on hit*, car nous voulons que ces particules disparaissent lorsqu'elles sont rattrapées par des particules prédatrices.

Maintenant, nous devons donner des règles à suivre aux particules. Dans le panneau *Physics*, choisissez *Boids* comme système de particules dans le menu déroulant, et modifiez les paramètres *Physics* de sorte que les proies soient un peu plus lentes que les prédateurs (afin de leur donner l'avantage de la vitesse sur les proies) en diminuant *MaxVelocity* à 7.000. Il n'est pas nécessaire de modifier les autres règles, car nous souhaitons mettre en place une simulation très simple.

### Les prédateurs

Maintenant, sélectionnez le cube *Predateurs*. Dans le menu *Object [F7]*, dans les *Particles buttons*, cliquez sur le bouton *Add New* du panneau *Particle System*. Changez quelques-unes des valeurs :

- *Amount* : 10 afin d'avoir une meute composée de 10 particules prédatrices ;
- *Life* : 1500 afin de donner à chaque particule une durée de vie suffisante pour que la simulation proies-prédateurs soit fonctionnelle.

Pour faciliter la compréhension visuelle de la simulation, dans le panneau *Visualization*, sélectionnez *Cross* à la place de *Point* et donnez aux particules une *Draw Size* de 10. Comme nous souhaitons que les prédateurs puissent aisément « dévorer » les particules proies, définissez leur taille *Size* à une valeur plus élevée, comme 1.5, ce qui signifie que dès qu'une particule proie pénétrera le rayon d'action d'une particule préda-

trice, elle sera dévorée. Parce que l'option *Die on hit* est activée pour la particule proie, celle-ci disparaîtra alors, tout simplement.

Maintenant, intéressons-nous à la partie la plus intéressante de cette simulation : dans le panneau *Physics*, choisissez *Boids* dans le menu déroulant, comme type de système de particules, et réglez son comportement (paramètres *Behaviour*) comme suit :

- *Goal* : 1.500 afin de favoriser la recherche des proies (objectifs) ;
- *Crowd* : 1.00 afin de défavoriser l'attroupement des prédateurs ;
- *Collision* : 0.500 ;
- *Avoid* : 0.500 ;
- tous les autres : 0.100-0.250.

Déplacez ensuite les propriétés *Goal* et *Crowd* plus haut dans la liste, en utilisant les petites flèches haut et bas en face de chaque règle, afin de les rendre prioritaires par rapport aux autres règles de même valeur.

#### BON À SAVOIR **Les règles comportementales des Boids**

Elles sont en nombre limité, mais permettent déjà de simuler un grand nombre de scénarios.

Elles poussent les *Boids* à :

- *Collision* : éviter les obstacles ;
- *Avoid* : éviter les prédateurs ;
- *Crowd* : éviter les autres boids ;
- *Center* : se placer au centre de l'essaim ;
- *AvVel* : maintenir la vitesse moyenne ;
- *Velocity* : calquer sa vitesse sur celle des boids voisins ;
- *Goal* : poursuivre son objectif ;
- *Level* : rester à la même altitude Z.

Des comportements normaux sont attendus avec des valeurs comprises entre 0.000 et 1.000. Mais les boids peuvent sur-réagir, conformément aux règles, avec des valeurs comprises entre 1.000 et 2.000. Réciproquement, avec des valeurs négatives comprises entre -1.000 et 0.000, le boid agira contrairement à la règle.

Il est également important de noter que l'ordre des règles dans la pile est important ; en effet, pour éviter un prédateur, un boid avec une forte valeur pour *Avoid* pourra accepter une collision malgré une forte valeur *Collision*, si *Avoid* est placée avant dans la pile.

Logiquement, les prédateurs sont doués pour la chasse. Afin de refléter ce fait, assurez-vous que leur vitesse maximale *MaxVelocity* (par défaut, 10.000) est plus grande que celle des proies (7.000 si vous avez suivi les instructions précédentes) et spécifiez une vitesse moyenne *AvVelocity* également plus élevée (par exemple, 0.500 alors que les proies ont une valeur de 0.300). De plus, toujours afin de conférer aux prédateurs des avantages stratégiques sur les proies, donnez-leur une meilleure accélération tangentielle *TangAcc* (par exemple, 0.500 alors que les proies ont une valeur de 0.200). Enfin, diminuez la taille de la meute à quelque chose comme *N* 3, de sorte que les prédateurs préfèrent chasser en petits groupes.

## Les Fields : comment fonctionnent-ils ?

Nous en avons pratiquement fini. Sélectionnez l'émetteur *Proies* et, dans le panneau *Fields* des *Physics buttons*, choisissez *Psys* à la place de *Object* et spécifiez-lui un field *Spherical* avec une force positive *Strength* égale à 2.000. Sélectionnez maintenant l'émetteur *Prédateurs* et, dans le même panneau *Fields*, sélectionnez *Psys.001* à la place d'*Object*. Comme précédemment, spécifiez-lui un field *Spherical*, mais cette fois avec une force négative *Strength* égale à -10.000 et une atténuation *Fall-off* de 2.000.

Pour expliquer les choses simplement, une valeur positive va dire à toutes les autres particules boids de la scène que ces particules sont des proies que les prédateurs trouveront appétissantes et tenteront d'attraper, grâce à la valeur *Strength* de 2.000 qui leur a été conférée. Dans le même ordre d'idée, la valeur négative va dire à tous les autres boids de la scène que ces particules sont très dangereuses, à cause de leur valeur *Strength* de -10.000 et que toutes les proies potentielles qui sont dans le secteur devraient absolument essayer de s'échapper de leurs griffes !

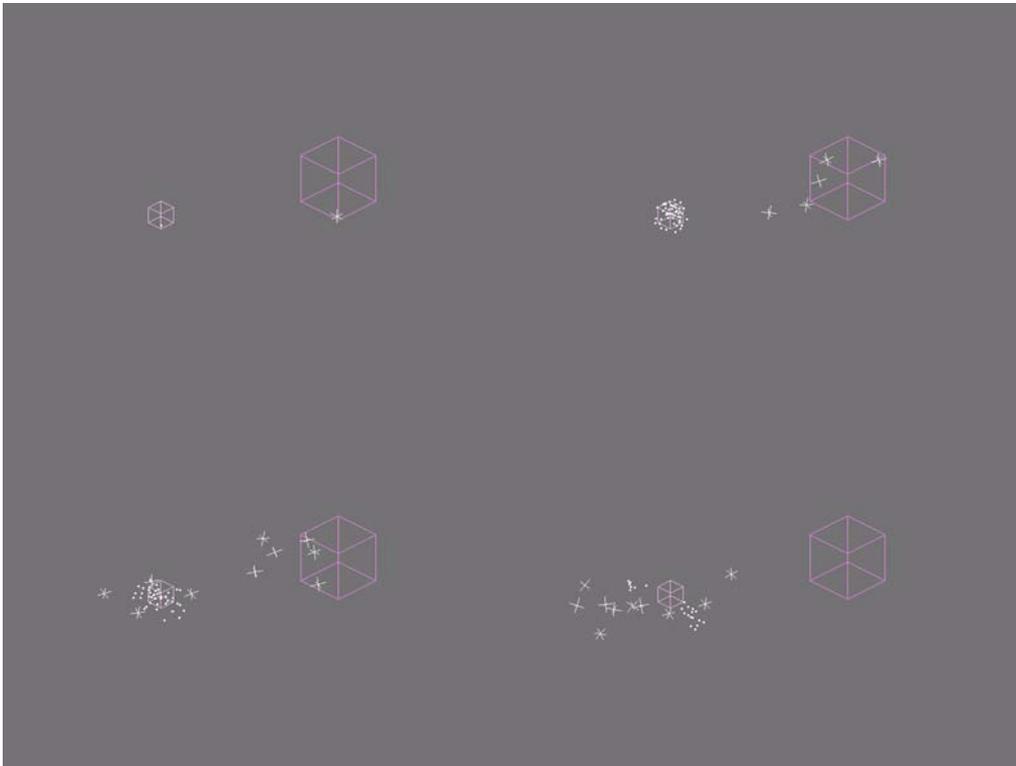
La valeur d'atténuation *Fall-off* fonctionne exactement comme pour les autres *Fields*. Par exemple, lorsqu'un prédateur est éloigné, la particule proie ne le remarquera pratiquement pas, car le *Fall-off* rendra le *Field* trop faible, passée une certaine distance. Mais plus le prédateur va s'approcher, plus l'effet du *Field* sera insistant, pressant la proie à fuir à toute volée dès que le prédateur se sera approché. Le paramètre *MaxDist* peut également être très utile ici, car grâce à lui, un prédateur peut s'approcher très près d'une proie avant d'être détecté ; lorsque la distance maximale spécifiée est atteinte, la proie devient alerte et tente de s'enfuir.

## Jouer la simulation

La dernière étape consiste à déterminer la durée de la simulation. Cela se passe simplement dans l'onglet *Bake* des *Particle buttons* : il faut uniquement spécifier une valeur suffisamment élevée pour le bouton *End* afin que la simulation entière soit calculée ; une valeur de 1000-1500 paraît cohérente vis à vis de la durée de vie *Life* spécifiée pour les particules, mais quelques essais permettront de s'en assurer. Ajustez cette valeur pour *chaque* système de boids mis en scène.

Si vous avez besoin d'ajuster la durée de vie *Life* des particules, pensez à libérer la mémoire de la simulation courante, en cliquant sur le bouton *Free Cache* de l'onglet *Bake*. Vous lancerez le calcul de la simulation grâce au bouton *Bake* du même onglet, ou en lançant l'animation grâce à *[ALT]+[A]* avec le curseur de la souris dans la vue 3D de votre choix. La touche *[ESC]* mettra fin à la simulation.

Vous retrouverez le résultat de cette simulation dans le fichier `exercice-07.09-final.blend`.



**Figure 7–82** Les prédateurs (croix) fondent sur les proies (points) et les déciment en quelques secondes de course-poursuite désespérée.

## Conclusion

Vous avez désormais toutes les cartes en main pour réaliser des animations assez sophistiquées, même si toutes les possibilités sont loin d'être couvertes. En fait, grâce à ces outils, vous pouvez virtuellement animer n'importe quoi, mais pour certaines animations particulièrement complexes, Blender propose d'autres outils, plus avancés mais aussi plus spécifiques. Cette spécialisation est le prix à payer pour obtenir des animations particulièrement réalistes, et impose à l'utilisateur l'acquisition de compétences supplémentaires.

chapitre

# 8



Copyright 2006 Jason Pierce. All rights reserved.

Tree frog and dragonfly, © 2006 par Jason Pierce : <http://cs.unm.edu/~sketch/gallery/>

# Techniques d'animation avancées

Ce chapitre vous montrera les techniques d'animation les plus avancées de Blender. Elles n'en sont pas pour autant forcément les plus complexes, mais les plus diversifiées et différentes du reste des autres fonctionnalités de Blender.

## SOMMAIRE

- ▶ Animation de corps souples
- ▶ Animation de corps rigides
- ▶ Simulation de systèmes fluides
- ▶ Animation squelettale
- ▶ Éditeur d'actions
- ▶ Éditeur d'actions non linéaires

## MOTS-CLÉS

- ▶ Softbody
- ▶ Cloth
- ▶ Goal
- ▶ Bake
- ▶ Game Engine
- ▶ Bullet
- ▶ Fluides
- ▶ Armature
- ▶ Squelette
- ▶ Rig
- ▶ Action
- ▶ Bandes d'actions
- ▶ Composition

### 3DS MAX Module d'extension Reactor

Dans 3ds max, les animations physiques sont introduites par un module d'extension spécifique qui permet la simulation de corps rigides, de corps souples, mais aussi de l'eau et du vent. Dans Blender, la première est héritée du *Game Engine*, la seconde d'un module de corps souples, la troisième d'un simulateur de fluides spécifique, et la dernière est un phénomène capable d'affecter aussi bien particules que corps souples, le tout sans avoir besoin de recourir à des extensions ou des scripts externes.

#### COMPRENDRE Théorie fondamentale des corps souples

Pour bien comprendre le fonctionnement d'un corps souple et, plus tard, mieux interpréter les paramètres permettant de régler la simulation, il suffit de remplacer, mentalement, chaque arête de votre maillage par un ressort et chaque point de contrôle par une petite masse de plomb. Ainsi, votre corps souple sera constitué d'une multitude de ressorts, chacun présentant la même rigidité. Il s'agit du paramètre *E Stiff* (rigidité de l'arête), dans l'onglet *Soft Body*.

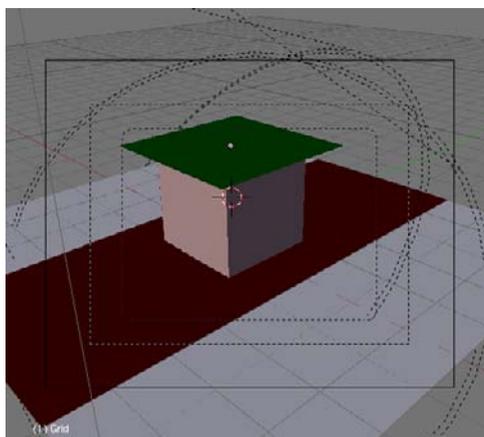
Le système d'animation de Blender a complètement été revisité par ses développeurs, au fil des mois, depuis la première édition de cet ouvrage. L'existant a été réécrit et amélioré (en particulier l'animation de personnages, désormais presque une formalité), tandis que des domaines bien plus avancés, tenant plus de la simulation physique que de l'animation, ont été ajoutés et améliorés ; on notera notamment l'arrivée des corps souples (idéaux pour l'animation de textiles ou de chevelures), celle plus balbutiante des corps rigides (tenant compte de la géométrie et de la masse des objets de vos scènes), ou encore la simulation des fluides. Ce chapitre vous propose de découvrir chacun de ces domaines.

## Les corps souples

Les corps souples (*Soft Bodies*, en version originale) constituent un système permettant de simuler la déformation d'objets en réponse à des sollicitations « extérieures ». Par exemple, un objet en acier écrasant un autre en mousse, le vent déplaçant des rideaux, les cheveux d'un coureur, etc. Outre la définition géométrique du corps souple à son état « au repos », il nous faudra définir sa rigidité globale. Plutôt que de présenter la théorie des corps souples, nous allons traiter deux exemples complets qui nous permettront de saisir la logique de fonctionnement.

### La nappe

Nous allons commencer par un exemple très simple de l'usage des corps souples, qui va nous permettre d'appréhender les limites du système, dans le cadre de l'animation de vêtements ou de textiles. Il va simplement s'agir d'appliquer une nappe sur un objet anguleux.



**Figure 8-1**

La scène de départ : prêts à passer à table ?

Ouvrez le fichier `exercice-ch08.01-depart.blend` du répertoire `/exercices` du DVD-Rom. L'écran principal se divise en deux parties : à gauche, une vue de face en mode fil de fer, et à droite, pour la visualisation de nos travaux, une vue depuis la caméra en mode ombré.

Normalement, la nappe est présélectionnée et l'on peut directement visualiser, dans le menu *Object* (touche [F7]) les panneaux relatifs au *Physics buttons*. Repérez l'onglet *Soft Body* et cliquez sur le bouton *Soft Body*.

L'onglet *Soft Body* est actualisé immédiatement pour présenter une collection de boutons qui peuvent paraître intimidants. Leurs fonctions sont toutefois assez simples à saisir. Ces boutons sont regroupés en trois blocs, de haut en bas : le premier permet de paramétrer les phénomènes externes au corps souple.

- *Mass* : il s'agit de la masse du corps souple. Pour un vêtement ou un textile, vous pouvez laisser à la valeur par défaut, égale à 1.000.
- *Grav* : il s'agit de la force de gravité qui s'exerce sur le corps souple ; celui-ci peut alors être mis en mouvement si la valeur est non nulle. Pour simuler la gravité terrestre, utilisez une valeur égale à 9.810.
- *Speed* : il s'agit d'un facteur de vitesse qui vous permet de corriger la vitesse de déformation du corps souple, si celle-ci n'est pas conforme à vos attentes. Pour démarrer, laissez la valeur initiale de 1.000, et corrigez la valeur en fonction de vos besoins. Par exemple, 2.000 pour doubler la vitesse, ou 0.500 pour la diminuer de moitié.

Le deuxième bloc de boutons permet de gérer les contraintes externes appliquées au corps souple. Par exemple, notre nappe pourrait tenir par un ou plusieurs coins, et il est assez simple de le simuler en utilisant l'option *Use Goal* de ce bloc. Toutefois, nous verrons plus tard un exemple d'utilisation de cette option. Dans l'immédiat, désactivez le bouton *Use Goal*.

Le troisième bloc de boutons, enfin, est celui qui permet de gérer l'essentiel du comportement du corps souple.

- *Use Edges* : en activant cette option, chaque arête du maillage est considérée comme un ressort, dont la rigidité est déterminée ci-après. Dans notre cas, prenons garde à laisser ce bouton actif. Si nous le désactivons, en cours de simulation, les arêtes pourront s'allonger à l'infini.
- *Pull* et *Push* : ces deux paramètres permettent de spécifier la raideur des ressorts aux arêtes, lorsque leur longueur est respectivement plus grande (*Pull*, force de rappel) ou plus courte (*Push*, force de poussée) que celle au repos. En d'autres termes, il s'agit d'une indication de l'effort fourni par le ressort de l'arête pour ramener celle-ci à sa longueur initiale. Des valeurs élevées induisent des systèmes très rigides et très dynamiques. Essayez une valeur *Pull* 0.250 et une valeur

## 3DS MAX Simulation de tissus

Dans 3ds max, il est fait usage d'assistants pour définir les objets qui constitueront des obstacles, le corps souple étant pour sa part défini sous forme de modificateur. Dans Blender, tout se passe dans les *Physics buttons* du menu *Object*, touche [F7]. Les propriétés d'obstacle des objets sont définies dans l'onglet *Collision*, tandis que celles de corps souples sont activées et réglées dans l'onglet *Soft Body*.

### Le corps souple en tant que modificateur

Une fois créé, un effet de corps souple apparaît automatiquement dans la liste des modificateurs de l'objet : menu *Editing* (touche [F9]), onglet *Modifiers*.

À ce titre, il profite des mêmes options que les autres modificateurs, et peut être déplacé dans la pile des modificateurs, activé ou non dans les vues 3D, activé ou non au moment du rendu, ou appliqué de façon permanente et non plus comme un modificateur.



Figure 8–2

### BON À SAVOIR Aérodynamisme

En activant le bouton *N*, vous prenez en compte, pour les corps souples, un comportement aérodynamique plus réaliste, le rendant plus sensible au vent (field *Wind*) et permettant de le faire flotter sous son influence. Commencez vos tests avec des valeurs *Aero* de l'ordre de 30–50 environ.

*Push* 0.650 pour avoir une nappe se déformant de façon assez satisfaisante. Des valeurs extrêmes induiront un comportement plutôt irréaliste avec des force de rappel ou de poussée chaotiques.

- *Stiff Quads* : activez ce bouton si vous constatez que le corps souple s'écrase trop sur lui-même. L'effet produit aidera à maintenir la cohésion de sa forme.
- *Damp* : permet de déterminer l'amortissement des ressorts. Plus la valeur est faible, plus les arêtes oscilleront au cours de la simulation. Au contraire, plus la valeur sera élevée, moins la longueur des arêtes oscillera. Dans notre exercice, une valeur de 5.000 sera satisfaisante.

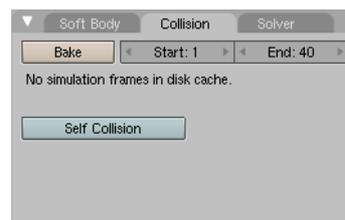


**Figure 8-3**  
Le panneau Soft Body de notre nappe

#### ASTUCE Gestion de l'autocollision

Les boutons *CEdge* et *CFace* permettent de vérifier, à chaque moment de la simulation, si les arêtes et les facettes (respectivement) entrent en collision. Si ces options sont idéales pour la simulation de tissus ou de corps souples grâce à une meilleure gestion des plis et des collisions, elles sont malheureusement plus intensives en temps de calcul et tendent à induire un « frissonnement » quasi permanent – même en l'absence de mouvements – qui ne peut être contré, pour *CEdge*, que par le paramètre *Damp*.

Le lancement de la simulation se fait grâce au bouton *Bake* de l'onglet *Collision* ou lorsque vous utilisez la combinaison *[Alt]+[A]*. Les boutons numériques *Start* et *End* déterminent les frames de départ et de fin de la simulation de corps souple. Nous choisirons respectivement 1 et 40.



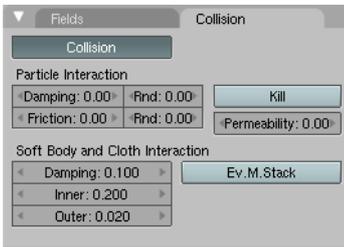
**Figure 8-4**  
Les paramètres de la simulation

#### COMPRENDRE Simulation précalculée

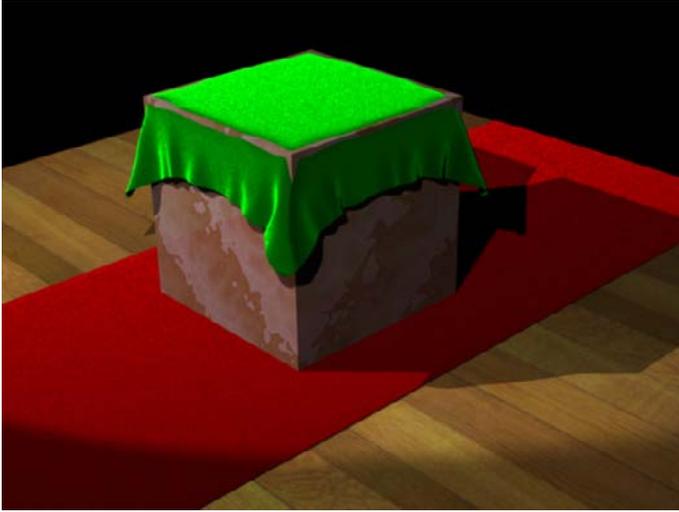
Dans Blender on ne parle pas de créer la simulation, mais de la calculer et de l'intégrer en tant qu'élément précalculé de l'animation. Cela se fait au travers du bouton *Bake*. Lorsque l'animation sera jouée, le corps souple se déformera toujours de la même façon, même si des éléments auront été retirés ou ajoutés. Il faudra alors cliquer sur le bouton *Free Bake* pour rejeter la simulation précédemment précalculée, et en relancer une nouvelle, toujours grâce au bouton *Bake*.

Il ne nous reste plus qu'à appuyer sur le bouton *Bake* et observer, dans la vue ombrée, la nappe tomber sur notre cube et le traverser ! Appuyez sur le bouton *Free Bake* pour rejeter cette solution.

En effet, aucune collision n'a été détectée. C'est parce qu'il faut définir le cube de notre scène comme étant un obstacle. Sélectionnez-le (*bouton droit* de la souris) et observez l'onglet *Collision*, toujours dans les *Physics buttons* du menu *Object* (touche *[F7]*). Activez le bouton *Collision*, ne touchez à aucun paramètre, et sélectionnez à nouveau notre nappe. Cliquez sur *Bake* et observez la nappe tomber sur le cube et l'envelopper ! Vous retrouverez, pour étude, le résultat de cette première partie de l'exemple dans le fichier `exercice-ch08.01-intermediaire.blend`.

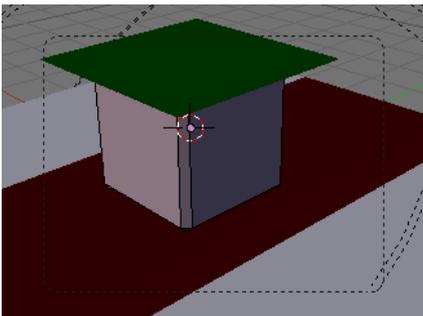


**Figure 8-5**  
Le panneau Collision du cube



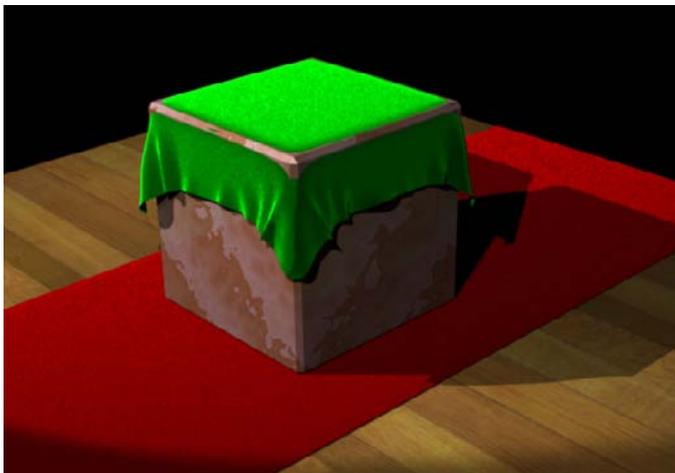
**Figure 8-6**  
Une simulation imparfaite,  
mais encourageante

Au vu du résultat, il apparaît assez clairement que le simulateur de corps souple a du mal avec les arêtes vives ! Nous allons essayer de voir comment corriger ce problème. Une première méthode va consister à chanfreiner le cube, de sorte qu'il ne présente plus d'arêtes aussi vives. Sélectionnez le cube (*bouton droit* de la souris), entrez en mode *Edit* (touche *[Tab]*), appuyez sur la touche *[W]* et, dans le menu flottant, choisissez *Bevel*. Déplacez la souris dans la vue 3D, jusqu'à ce que le chanfrein de la taille souhaitée apparaisse.



**Figure 8-7**  
Le cube et ses arêtes chanfreinées

Quittez le mode *Edit* et sélectionnez à nouveau la nappe en suspension, cliquez sur *Free Bake* pour rejeter la simulation précédente, et cliquez à nouveau sur *Bake*.

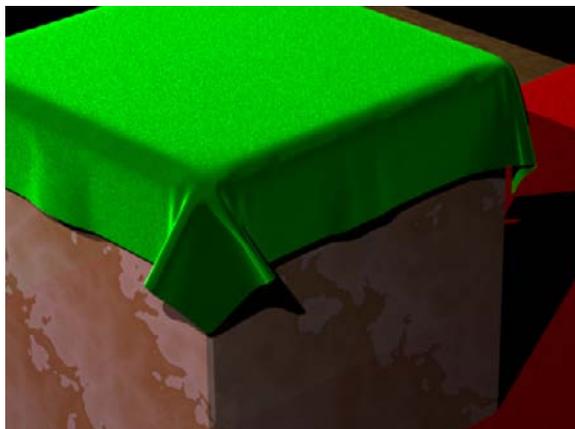


**Figure 8-8**  
Le résultat est meilleur, mais pas encore assez bon.

Non seulement le résultat n'est pas fondamentalement meilleur, mais en plus vous ne souhaitez pas toujours chanfreiner vos objets.

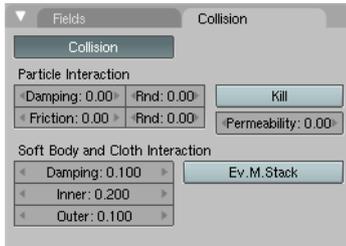
**BON À SAVOIR Les meilleurs maillages pour une bonne simulation**

En règle générale, les formes progressives arrondies seront plus facilement recouvertes par des corps souples, mais le seul moyen d'améliorer la qualité de la simulation consistera, malheureusement, à subdiviser le maillage du corps souple de sorte que sa plus grande face soit plus petite que la plus petite des faces de l'objet recouvert. Du point de vue temps de calcul, cette approche (bien que correcte et rigoureuse) devient vite catastrophique !



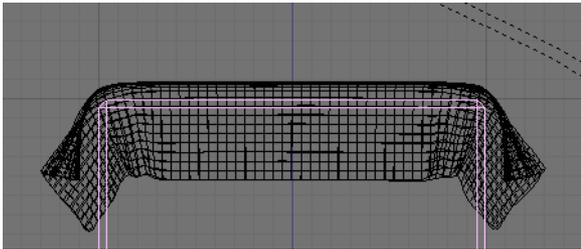
**Figure 8-9** Exemple avec un cube chanfreiné et une nappe très subdivisée !

Heureusement, il y a au niveau de l'obstacle (le cube) des réglages qui nous aideront à tricher : en n'y regardant pas de trop près, nous obtiendrons le résultat visuel souhaité. Par exemple, augmentez le paramètre *Outer* à 0.100.



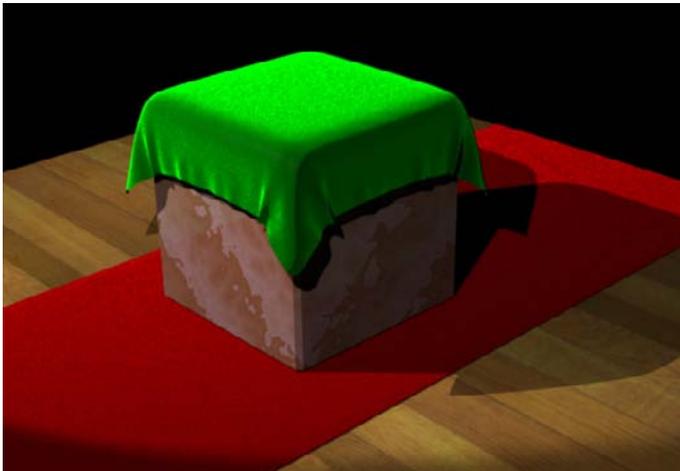
**Figure 8–10**  
Le paramètre de déflexion *Outer*

Ce faisant, vous imposez à votre objet une surépaisseur virtuelle, qui va aider le corps souple à l'envelopper proprement. Bien sûr, si vous zoomez en mode fil de fer, vous apercevrez le corps souple flotter à 0.100 unités au-dessus du cube.



**Figure 8–11**  
Le corps souple ne repose pas sur le cube.

Mais le résultat est visuellement atteint, puisque avec notre angle de caméra, ce décalage n'est pas observable. Vous pouvez jouer avec le fichier final de cet exemple, qui se nomme `exercice-ch08.01-final.blend` et qui se trouve dans le répertoire `/exercices` du DVD-Rom.



**Figure 8–12**  
Notre nappe enfin posée !

#### MÉTHODE Le simulateur de corps souple : un outil de modélisation ?

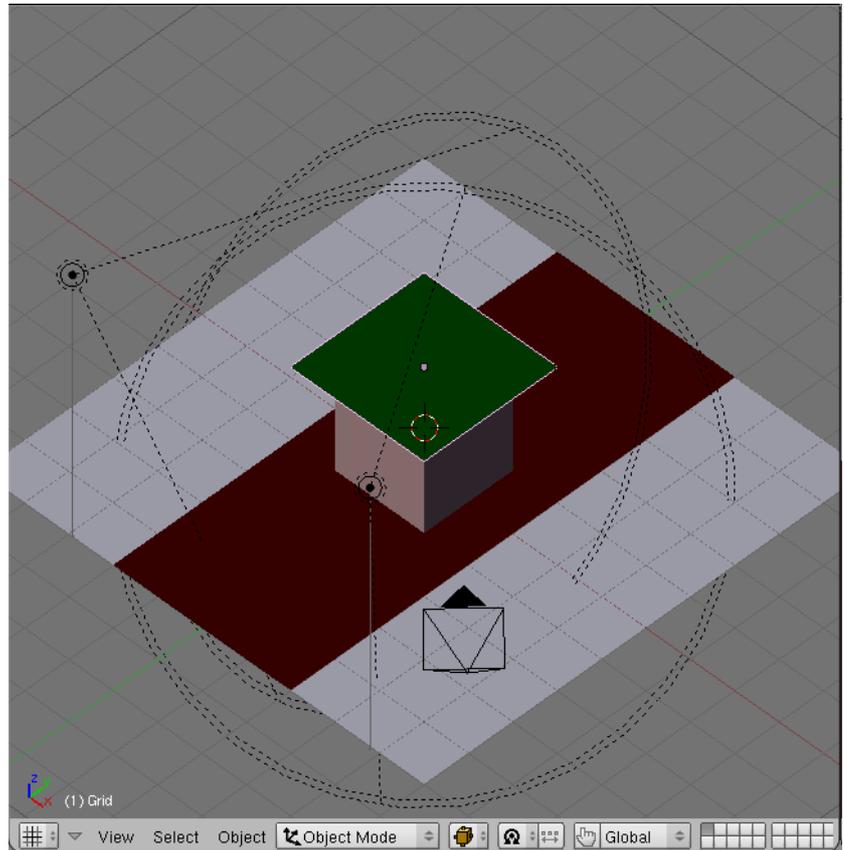
Nous vous avons déjà fait remarquer que les corps souples apparaissent dans la liste des modificateurs sous le nom de *Softbody*. Si vous cliquez sur le bouton *Apply*, la déformation est appliquée au maillage de façon permanente. Il ne peut, dès lors, plus être animé comme un corps souple, mais a sa géométrie figée, comme si vous l'aviez modélisé en l'état.

## Fixer une partie du corps souple

Nous allons renforcer notre connaissance des corps souples en découvrant une application possible des boutons *Use Goal*, grâce au *Weight Painting*.

Qu'est-ce qu'un *Goal* ? Lorsque vous spécifiez un *Goal*, en particulier sur un groupe de points de contrôle prédéfinis, c'est comme si vous les clouiez sur place. Si la valeur *Goal* est égale à 1.0, l'objet se comportera comme n'importe quel autre objet, sans effet de corps souple ; il est alors normalement affecté par les courbes IPO, les modificateurs de déformation, les parents et autres. En revanche, si la valeur *Goal* est égale à 0.0, l'objet n'est affecté que par les lois physiques du système de corps souple.

Le *Goal* permet aussi d'assurer la mémoire de forme. Il servira à vérifier que l'objet ne se déformera pas trop, et représente en quelque sorte la « mémoire de forme » de l'objet. Grâce au système de *Weight Painting*, vous pouvez assigner des valeurs de *Goal* différentes pour les points de contrôle de l'objet.

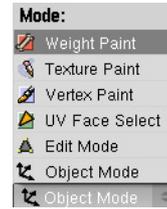


**Figure 8-13**  
Notre scène de départ, déjà familière

Ouvrez le fichier `exercice-ch08.02-depart.blend` du répertoire `/exercices` du DVD-Rom. Dans la vue 3D à gauche, passez en vue de dessus (touche [7] du pavé numérique) et, avec la nappe sélectionnée, cliquez sur le bouton *Mode* pour afficher les modes de représentation possibles. Choisissez *Weight Paint*.

La nappe apparaît désormais en bleu uni, malgré que l'on soit en mode fil de fer et que le matériau défini pour la nappe ne soit pas du tout bleu. C'est parce que nous sommes entrés dans un menu spécial qui nous permettra de pondérer les zones à déformer : les couleurs froides (le bleu foncé étant à l'extrême) indiqueront les zones totalement libres d'être déformées ; au contraire, les couleurs chaudes (le rouge étant à l'extrême) indiqueront les zones qui ne se déformeront pas. Dans notre cas, nous souhaitons que la nappe entière soit libre de bouger (bleu foncé, valeur *Weight* 0.000) à l'exception de deux coins (rouge, valeur *Weight* 1.000) qui pourraient être tendus par des mains. Nous allons donc essayer de peindre notre nappe de façon à prendre en compte ces degrés de liberté.

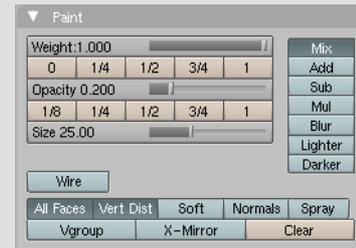
Vous aurez noté, au passage, qu'en entrant dans le mode *Weight Paint*, le curseur de la souris a changé de forme pour représenter un petit pinceau et qu'un nouveau panneau, nommé *Paint*, a fait son apparition dans le menu *Editing* (touche [F9]).



**Figure 8–14**  
Choix du mode *Weight Paint*

### Le mode *Weight Paint*

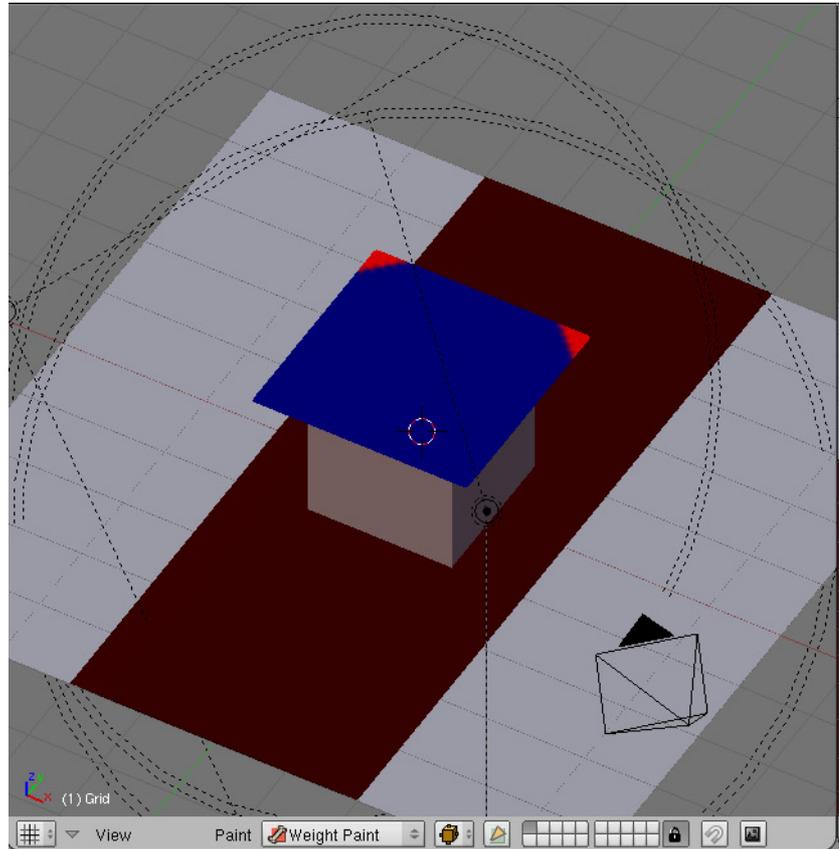
- Le curseur *Weight* permet de déterminer la couleur (ou chaleur) de la zone sous le pinceau. *Weight* 0.000 correspond à la couleur la plus froide (le bleu foncé) et *Weight* 1.000 à la couleur la plus chaude (le rouge vif). Vous noterez quelques valeurs pré-réglées du curseur (il vous suffit de cliquer dessus pour régler le *Weight* à la valeur indiquée) : 0, 1/4, 1/2, 3/4 et 1.
- Le curseur *Opacity* définit l'opacité de la « peinture » appliquée. Une valeur de 0.000 ne change pas la couleur du rideau tandis qu'une valeur de 1.000 applique directement la couleur imposée par le curseur *Weight*. Toute valeur intermédiaire permet de superposer la portion de *Weight* déterminée par l'*Opacity* à l'objet, selon des règles prédéterminées (*Mix*, *Add*, *Sub*, *Mul*, *Filter*, *Lighter*, *Darker*). À noter également des valeurs de curseur pré-réglées : 1/8, 1/4, 1/2, 3/4 et 1.
- Le curseur *Size* permet de définir la taille du pinceau utilisé.



**Figure 8–15** Le panneau *Paint*

La valeur *Weight* est normalement déjà égale à 1.000. En revanche, *Opacity* est à 0.200. Réglez cette valeur à 1.000, directement avec le curseur ou en cliquant sur le bouton intitulé 1, juste en dessous de ce paramètre. Puis, avec le bouton gauche de la souris, peignez en rouge vif deux des coins de la nappe. Le bouton *X-Mirror* permet de ne peindre qu'un seul angle, et laissez Blender reproduire de façon symétrique les coups de pinceau. Éventuellement, le bouton *Wire* peut être activé pour visualiser le maillage en cours de peinture.

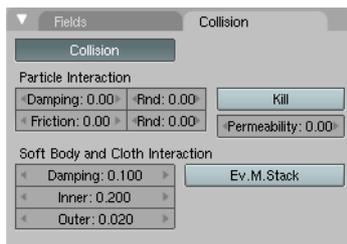
**Figure 8-16**  
La nappe en mode Weight Paint



#### ASTUCE

### Modifier le Weight par petites touches

Pour procéder progressivement (un peu comme si vous peigniez à l'aérographe), il vous suffit de fixer *Weight* à 0.000 (pour « refroidir » les couleurs) ou à 1.000 (pour les « réchauffer ») et de jouer sur la valeur *Opacity*. Vous pouvez également activer l'option *Spray* pour augmenter progressivement la valeur du *Weight*.



**Figure 8-17**  
Le panneau Collision du cube

Maintenant, l'ensemble de la nappe est sujette à déformation totale (couleur bleue foncée) à l'exception deux coins (couleur rouge). Nous pouvons maintenant nous intéresser à la simulation du corps souple à proprement parler.

Revenez dans le mode *Object Mode*. Avec le bouton droit de la souris, sélectionnez le cube sous la nappe. Allez ensuite dans le menu *Object* (touche [F7]) puis dans le panneau *Collision* pour activer le bouton *Collision*.

Désormais, le cube est considéré comme un obstacle qui pourra stopper la nappe au lieu de la laisser le traverser, comme un fantôme. Mais, pour obtenir ce résultat, il nous reste encore à définir les propriétés de corps souple de la nappe.

Sélectionnez la nappe et dans les *Physics buttons* du menu *Object* (touche [F7]), activez le bouton *Soft Body* de l'onglet *Soft Body*. Dans le premier bloc de boutons, spécifiez une valeur pour *Grav* de 9.810. Dans le deuxième bloc, vérifiez que le bouton *Use Goal* est actif et choisissez dans le menu déroulant le groupe de points de contrôle créé grâce au

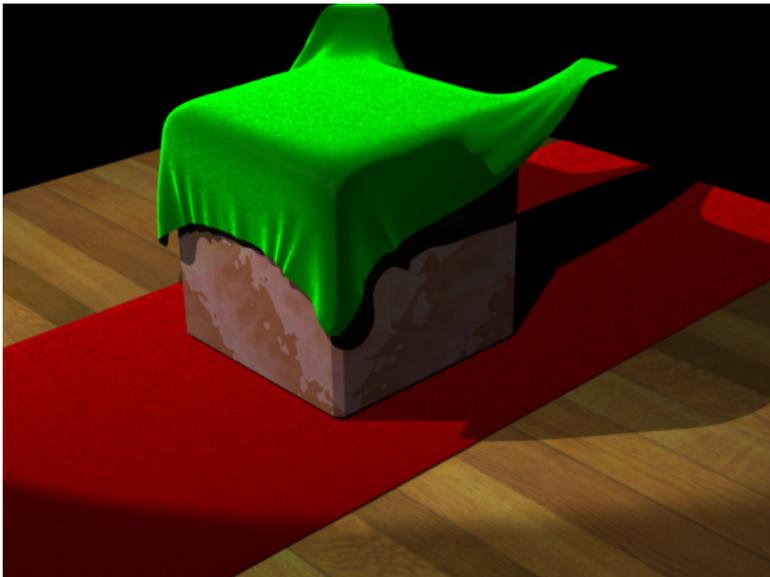
*Weight Painting*. Ces opérations sont incontournables pour prendre en compte le *Weight Painting* réalisé précédemment. Spécifiez une valeur *G Stiff* égale à 0.000 pour indiquer que la raideur globale de la nappe est nulle. Dans le troisième bloc, vérifiez que *Use Edges* est actif et activez l'option *Stiff Quads* ; changez les valeurs *Pull* à 0.250 et *Push* à 0.650. Spécifiez également *Damp* 5.000 pour limiter les balancements de la nappe.



**Figure 8–18**  
Le panneau Soft Body

Affichez maintenant l'onglet *Collision* pour afficher les paramètres d'animation du corps souple. Spécifiez une simulation comprise entre les frames 1 (*Start*) et 50 (*End*) et cliquez ensuite sur le bouton *Bake* pour lancer la simulation et en enregistrer le résultat sur le disque.

Bien évidemment, la simulation est tout aussi imparfaite que précédemment et requiert de chanfreiner le cube et de jouer sur son paramètre *Outer* pour que tout soit parfait. Vous pouvez consulter le résultat final dans le fichier `exercice-ch08.02-final.blend` du répertoire `/exercices` du DVD-Rom.



**Figure 8–19**  
Des mains invisibles maintiennent la nappe au-dessus du cube.

### Les paramètres G Min et G Max

Ce sont deux paramètres qui permettent de changer les bornes du *Weight Painting* en redéfinissant la valeur de *Goal* des couleurs froides (*G Min*, par défaut 0.000) et des couleurs chaudes (*G Max*, par défaut 1.000). Par exemple, avec *G Min* 0.400 et *G Max* 0.85, la couleur la plus froide du *Weight Painting* sera équivalente à un *Goal* de 0.40 et la couleur la plus chaude à un *Goal* de 0.85. Aucun point du corps souple n'est alors totalement inamovible, mais aucun n'est non plus totalement soumis aux lois de la physique.

### ASTUCE Correction de la simulation

Lorsque la simulation ne vous satisfait pas totalement, vous pouvez activer l'option *Bake Editing*, et vous positionner sur la *frame* dont le résultat vous déplaît. Tout à fait normalement, passez en mode *Edit* (touche *[Tab]*) et corrigez le maillage du corps souple pour cet instant. Quittez le mode *Edit* et appuyez sur le bouton *Rebake From Current Frame* pour reprendre la simulation à partir de la frame courante.

## Réglages du moteur de corps souple

Il est possible de régler individuellement, pour chaque corps souple, les paramètres relatifs aux collisions dans l'onglet *Collision*. Les paramètres proposés permettent de simuler l'autocollision des corps souples, ce qui est idéal pour la simulation de vêtements ou de textiles, en permettant d'obtenir des plis plus réalistes.

En activant le paramètre *Self Collision*, le solveur positionne à chaque sommet une balle viscoélastique fictive qui permettra de prendre en compte l'autocollision du corps souple, ce qui est particulièrement utile pour la simulation de tissus ou de draps. Il est ainsi possible de spécifier une dimension manuelle (option *Man*) au travers du paramètre *Ball Size* ou de choisir un modèle de simulation parmi *Av*, *Min*, *Max* et *AvMiMa* qui prennent en compte la longueur des arêtes/ressorts. L'option *Min* peut contribuer à laisser passer un pli du corps souple au travers d'un autre, tandis que *Max* se révèle plus fiable mais aussi plus long à calculer. *B Stiff* détermine dans ce cadre la rigidité des balles viscoélastiques et *B Damp* leur coefficient de frottement.

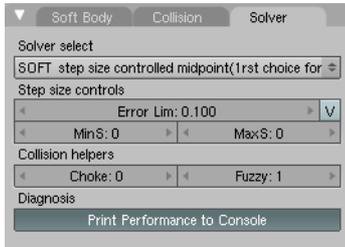


Figure 8–20  
L'onglet Collision

Il est également possible de régler les paramètres généraux de la simulation de corps souple grâce à l'onglet *Solver*. Tout d'abord, il est possible de choisir le solveur de la simulation : *SOFT* est la méthode à retenir, mais *RKCP* produit des résultats plus conformes aux lois physiques ; il s'agit toutefois d'une méthode bien plus difficile à paramétrer. Parmi les paramètres disponibles, *Error Lim* est un critère de convergence (méthode *Runge-Kutta*), des valeurs plus faibles que le défaut permettant une plus grande précision de la simulation, au prix d'un temps de calcul supérieur. Si une simulation ne paraît pas satisfaisante, il est possible d'imposer un nombre minimal et maximal de pas de calcul pour chaque frame (*MinS* et *MaxS*).

En augmentant la valeur du paramètre *Fuzzy*, vous accélérez le calcul de la simulation, mais au prix d'un calcul plus approximatif des collisions. Au contraire, si les réactions du corps souple à la suite d'une collision sont disproportionnées ou si le corps souple tend à passer à travers l'obstacle au lieu de glisser sur lui, vous pouvez augmenter le paramètre *Choke* pour améliorer la simulation.

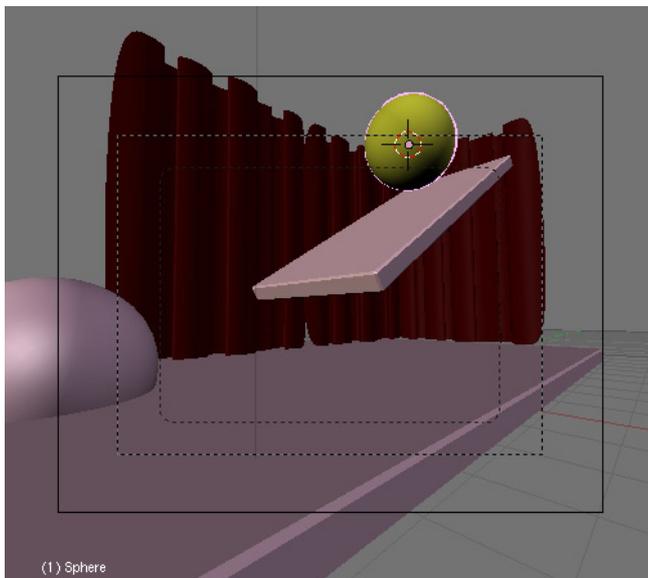
Enfin, l'option *Print Performance to Console* permet d'afficher dans la console de Blender quantité d'informations pertinentes quant à l'état de la simulation. Notamment, le nombre d'étapes de calcul par frame est précisé, de même que le temps moyen de calcul pour chacune d'elles.



**Figure 8–21**  
L'onglet Solver

## La balle en mousse

L'animation physique est un domaine où l'on n'attend pas forcément le simulateur de corps souple. Pourtant, grâce à la gravité et aux obstacles, il est facile d'organiser une animation mettant en jeu une balle qui tombe sur un plan incliné, puis frappe un obstacle avant de rouler au sol. C'est ce type d'animation que nous allons maintenant, très simplement, mettre en œuvre. Pour renforcer le lien avec le sujet des corps souples, la balle sera en mousse, et se déformera à chaque impact.



**Figure 8–22**  
Le parcours du combattant pour une balle en mousse

Ouvrez le fichier `exercice-ch08.03-depart.blend` du répertoire `/exercices` du DVD-Rom. Son écran principal est divisé en deux parties : une vue de dessus en mode filaire, avec une sphère présélectionnée, à gauche ; et une vue depuis la caméra, en mode ombré, sur la droite.

### BON À SAVOIR Résistance des matériaux

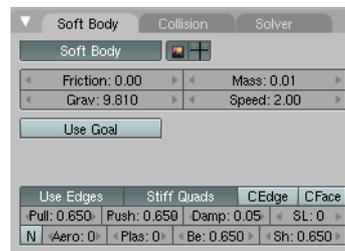
Lorsqu'ils sont fortement étirés ou comprimés, les matériaux sont généralement dégradés et subissent des déformations permanentes dites « plastiques ». De même, certains matériaux sont plus résistants à certaines sollicitations que d'autres : par exemple, plier une barre en fer pleine sera plus difficile que de plier un tube creux de mêmes dimensions. Les amateurs de simulations mécaniques apprécieront les paramètres suivants qui facilitent la simulation de caractéristiques mécaniques précises.

- *Plas* : *plastic deformation* suggère le taux de déformation permanente.
- *Be* : *bending stiffness* suggère la rigidité en flexion du corps souple.
- *She* : *shearing stiffness* suggère la résistance au cisaillement du corps souple.

### ASTUCE L'option Stiff Quads

Parfois, un objet a tendance à s'écraser trop facilement, en particulier sous l'effet de la gravité. L'usage de l'option *Stiff Quads* permet d'ajouter des ressorts supplémentaires virtuels sur la diagonale des faces quadrangulaires des maillages et donc de les aider à conserver cette forme.

**Figure 8-23**  
Les paramètres de corps souple de la balle en mousse



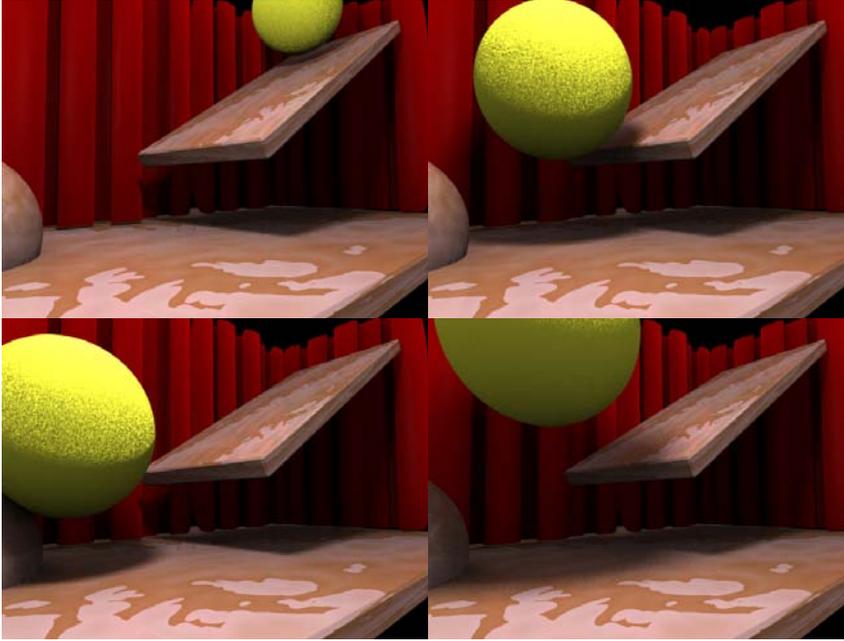
Dans la vue de dessus, sélectionnez le plan incliné ; dans l'onglet *Collision* des *Physics buttons* du menu *Object* (touche [F7]), activez le bouton *Collision*, mais ne touchez à aucun autre paramètre. Sélectionnez ensuite le sol, puis la sphère qui fait obstacle, et à chaque fois, activez de la même manière le bouton *Collision*.

Sélectionnez maintenant la balle en mousse. Dans l'onglet *Soft Body*, activez le bouton *Soft Body* et réglez ses propriétés comme suit :

- *Friction* : 0.000 ; en raison de la dureté des surfaces et en considérant la friction de l'air comme négligeable pour la simulation, on peut considérer qu'aucun frottement ne viendra perturber les mouvements de la balle ;
- *Mass* : 0.010 (soit 10 g) est une valeur bien suffisante pour une balle en mousse ;
- *Grav* : 9.810, comme dans les exercices précédents ;
- *Speed* : à vous de voir lors de vos simulations, mais une valeur de 2.000 semble donner le dynamisme souhaité pour ce genre de simulation ;
- *Use Goal* : cette propriété est bien évidemment désactivée ;
- *Use Edges* et *Stiff Quads* : propriétés activées, pour que la géométrie de la balle communique toute sa rigidité à la simulation ;
- *Pull* et *Push* : 0.65, afin d'avoir un matériau relativement élastique répondant rapidement aux sollicitations externes (écrasement, etc.) ;
- *Damp* : 0.05 pour avoir un minimum de dissipation d'énergie au cœur de la balle de mousse ; toute faible valeur peut également faire l'affaire ;
- *Be* : 0.65, pour que la balle ait une certaine rigidité et ne s'effondre pas sur elle-même comme un ballon de baudruche dégonflé ;
- *Sh* : 0.65, car la mousse n'offre pas une grande résistance mécanique au cisaillement.

Il ne nous reste maintenant plus qu'à configurer l'animation et lancer le calcul de la simulation. Pour cela, affichez l'onglet *Collision*. Après quelques tâtonnements, vous trouverez qu'un calcul entre la frame 1 (*Start*) et la frame 60 (*End*) sera amplement suffisant (avec *Speed* égal à 2.000). Cliquez alors sur le bouton *Bake* et observez, au fur et à mesure des cal-

culs, dans la vue depuis la caméra (en mode ombré), l'évolution de la balle et de ses déformations. Comme d'habitude, vous pourrez directement passer aux conclusions et ouvrir le fichier `exercice-ch08.03-final.blend`.



## Les corps rigides

Le système d'animation des corps rigides s'appuie sur la physique pour déterminer la trajectoire des objets, plutôt que sur des courbes IPO artificiellement imposées par l'utilisateur. C'est donc un système d'animation très réaliste, mais en même temps, un peu imprévisible. Malheureusement, son usage n'est (pour l'instant) prévu que dans le cadre du moteur de jeu en temps réel embarqué dans Blender. Toutefois, Blender propose d'enregistrer, sous forme de courbes IPO, les déplacements et rotations des objets impliqués dans la simulation de corps rigides, et donc de recycler cette simulation de façon classique dans Blender. Il est bien sûr alors possible d'éditer individuellement chaque courbe IPO, et donc d'isoler un objet et de le contrôler plus précisément.

La démarche de la simulation de corps rigides fait appel à des notions atypiques, car basées sur les mécanismes du moteur de jeu. Toutefois, nous verrons rapidement les bases qui nous sont nécessaires pour la mettre en œuvre dans des animations classiques.

### Pourquoi utiliser l'option Bake ?

Vous pouvez très bien vous en passer, mais une simulation qui l'a utilisée pourra être rejouée beaucoup plus rapidement, n'étant plus simulée en temps réel. De plus, si vous effectuez le rendu de votre animation avec l'option *Motion Blur* (ou *Vector Blur*) active, le résultat sera conforme à vos attentes, ce qui ne sera pas le cas autrement.

### Le bouton Apply Deformation

Grâce à ce bouton, Blender fige les déformations éventuelles du corps souple (par exemple, *Armature* ou *Lattice*) avant d'y appliquer la simulation.

**Figure 8-24**

La balle chute seule, se déforme à l'impact, et rebondit grâce au système de simulation des corps souples.

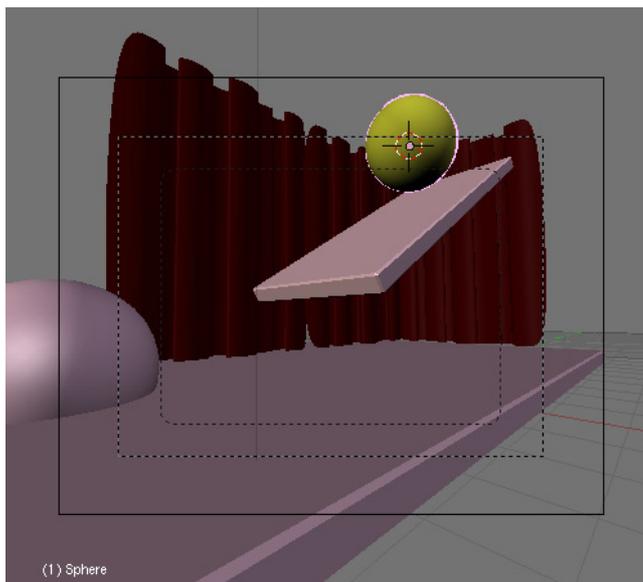
**MÉTHODE Choisir entre corps souples et corps rigides**

Ces deux systèmes d'animation peuvent tenir compte de la masse, de la gravité et des collisions avec l'environnement, pour déterminer les mouvements des objets concernés. Toutefois, chacun présente des spécificités qui les réserveront à un usage plutôt qu'à un autre.

Le système de corps souples tient bien évidemment compte de la déformation des objets impliqués. La géométrie de chaque corps souple étant réévaluée à chaque instant, les calculs sont relativement lourds et longs. Il est possible de faire interagir plusieurs corps souples entre eux, mais au prix de temps de calcul d'autant plus longs. En revanche, les corps souples ne sont pas pris en compte dans les simulations fluides.

Pour sa part, le système de corps rigides ne tient pas (par définition) compte de la déformation des objets. Il excelle en revanche dans l'animation de nombreux corps rigides, lâchés en même temps dans une simulation. Le calcul de l'animation est relativement rapide (le moteur est prévu pour du temps réel) pour peu que le nombre de facettes ne soit pas trop important. L'avantage est que les simulations fluides sont capables de tenir compte du résultat d'une simulation de corps rigides, mais a posteriori (pas en temps réel).

Enfin, les animations suivant ces trois systèmes de simulation (corps souples, corps rigides, fluides) ne sont malheureusement pas calculées lors de la même passe, mais individuellement, chacun avec son propre système de simulation. Si elles sont limitées par leur nature, les interactions sont toutefois possibles, avec un peu d'inventivité.

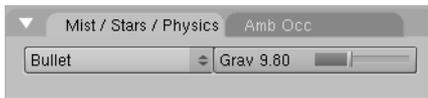


**Figure 8-25**  
Un terrain d'expérimentations  
identique au précédent

Ouvrez le fichier `exercice-ch08.04-depart.blend` du répertoire `/exercices` du DVD-Rom. La scène vous est certainement familière,

puisqu'il s'agit de la même que celle illustrant le système de corps souples. L'écran principal est séparé en deux parties : à gauche, l'éditeur de courbes IPO (pour l'instant vide), et à droite, une vue de la scène depuis la caméra, en mode fil de fer. La balle est sélectionnée.

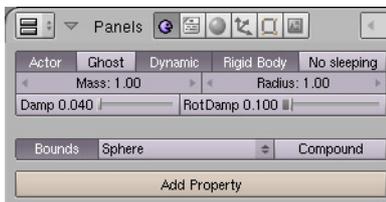
Commençons par spécifier à Blender le système de simulation physique que nous allons employer. Dans les *World buttons* du menu *Shading* (touche [F5]), repérez l'onglet *Mist/Stars/Physics*. Par défaut, le système employé doit être le *Bullet*, vérifiez que ce soit bien le cas. Sinon, changez-le de *Sumo* à *Bullet* en cliquant simplement sur le menu déroulant. Notez la présence d'un bouton *Grav* (valeur par défaut 9.80, qui nous convient parfaitement), qui nous permet de déterminer la gravité du système de simulation.



**Figure 8–26**  
Les paramètres de la simulation dans l'onglet Physics

Dans le menu *Logic* (touche [F4]), activez le bouton *Actor*, indiquant que le moteur de jeu va s'intéresser à l'objet sélectionné en particulier. Deux nouveaux boutons apparaissent. Activez *Dynamic* pour indiquer que cet objet devra être mis en mouvement par le moteur. Pour ce faire, tout un bloc de nouveaux boutons est présenté à son tour. Nous ne chercherons pas à produire une animation très sophistiquée, et nous laisserons donc la plupart des paramètres à leur valeur par défaut. Activez toutefois les boutons *Rigid Body* (c'est après tout le type d'animation que nous souhaitons pratiquer) et *Bounds*.

Cette méthode *Bounds* conditionne l'algorithme de détection des collisions du système. Par défaut, grâce à la valeur *Box*, il prend en compte la boîte minimale englobant l'objet animé, quelle que soit sa forme. Une collision est décrétée lorsqu'un quelconque objet de la scène cherche à pénétrer cette limite. Bien sûr, *Box* n'est peut-être pas le type de frontière le plus adapté à la forme de votre objet animé, peut-être très complexe. C'est pourquoi le système propose plusieurs types de primitives : *Box*, *Sphere*, *Cylinder*, *Cone*, *Convex Hull Polytope*, *Static triangle Mesh*.



**Figure 8–27**  
Les paramètres logiques de la balle à animer

### Sumo contre Bullet

Ce sont deux systèmes de simulation physique développés pour le moteur de jeu de Blender. Le système le plus ancien est *Sumo* ; il intègre parfaitement toutes les briques logiques nécessaires à la création d'un jeu. Du point de vue de la détection de collision, la frontière (*Bound*) de type *Sphere* est la seule à fonctionner admirablement, ce qui limite son utilisation pour des animations réalistes.

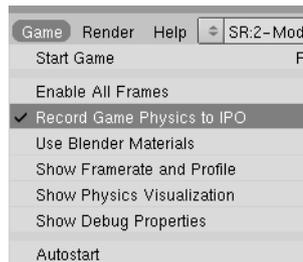
Le système *Bullet* est, pour sa part, en cours de développement. De par sa jeunesse, il est moins complet que son prédécesseur au niveau des briques logiques, mais offre une détection de collision supérieure avec les autres types de frontières. À moins d'être intéressé par la création de jeux vidéos avec Blender, concentrez-vous sur le système *Bullet*, le plus proche de nos besoins d'apprentis animateurs.

### OUPS ! Mauvaise détection des collisions

En raison d'un bogue connu de la version 2.40 de Blender, vos objets entrant en collision peuvent réagir de façon très brusque et totalement inattendue. Si vous rencontrez ce problème, sélectionnez un à un les objets devant entrer en collision, et utilisez la combinaison de touches **[Alt]+[R]** pour réinitialiser leur rotation (*Clear Rotation*). Utilisez ensuite la combinaison de touches **[Ctrl]+[A]** pour appliquer les échelles et les rotations, (*Apply Scale and Rotation*), puis repositionnez les objets à la main. Ils sont enfin prêts à être animés !

**Figure 8-28**

Activation de l'enregistrement de l'animation du jeu dans des courbes IPO



**Figure 8-29**

Plage d'enregistrement dans les courbes IPO



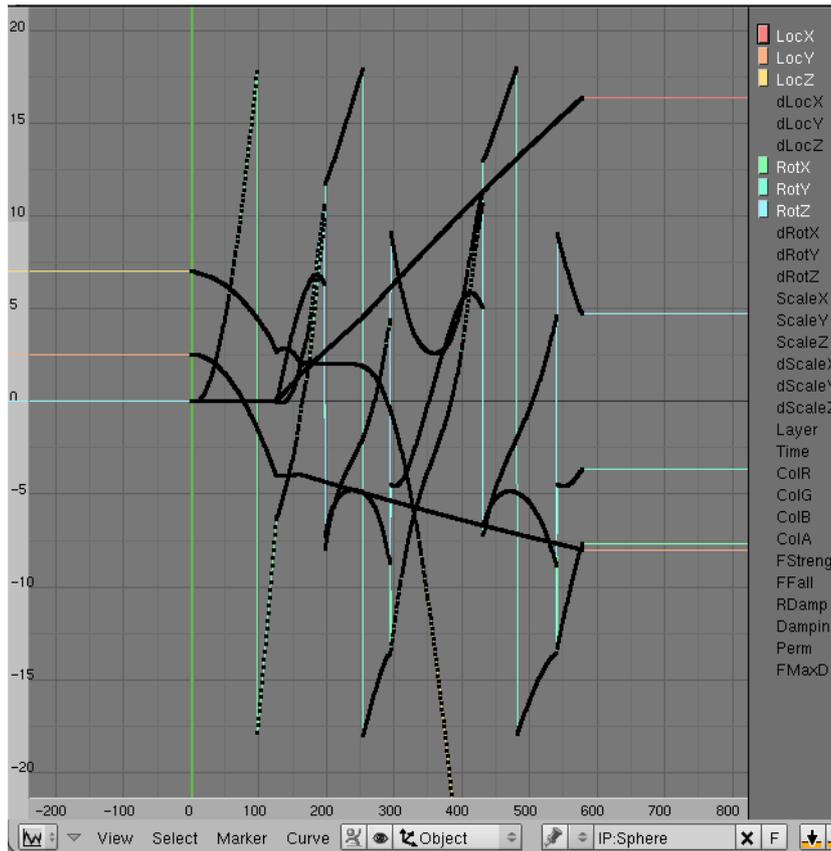
*Bounds* détermine l'espace inviolable de l'objet. Si un autre objet de la scène doit pénétrer celui-ci, il y a collision, et le moteur de simulation réagit, en fonction du mouvement initial, pour empêcher les objets de pénétrer l'un à l'intérieur de l'autre. Il y a plusieurs formes de frontières, mais l'une d'elles est tout indiquée en raison de la forme de notre balle : *Sphere* ! Dans la vue 3D, si vous zoomez sur la balle, vous remarquerez une sphère dont l'intérieur est dessiné en pointillés. Il s'agit de la frontière « physique » et il faut essayer de l'ajuster au plus près des dimensions de notre balle. Cela se fait grâce au bouton numérique *Radius* ; essayez différentes valeurs, mais nous estimons qu'une valeur de 1.400 est suffisamment correcte dans notre cas.

Dans le menu *Game* de la barre de menus principale, tout en haut de l'écran, activez l'option *Record Game Physics to IPO*. Cela aura pour effet (et c'est là tout l'intérêt de ce chapitre) d'enregistrer tous les mouvements (déplacements et rotations incluses) de notre balle dans des courbes IPO liées aux paramètres appropriés (*LocX* à *LocZ* et *RotX* à *RotZ*).

Les courbes ne seront toutefois pas enregistrées à l'infini, mais uniquement sur la plage d'animation spécifiée dans le panneau *Anim* des *Render buttons* du menu *Scene* (touche **[F10]**). Par défaut, ce sera donc enregistré entre les frames 1 (*Start*) et 250 (*End*).

Mais il nous reste encore à lancer le « jeu » et donc le système de simulation physique. Pour cela, vous pouvez, soit passer par le menu *Game* de la barre de menus principale (en choisissant *Start Game*), soit presser la

touche *[P]* avec le curseur de la souris, dans la vue de votre choix (de préférence, celle affichant la vue depuis la caméra). Attendez patiemment que la boule descende, rebondisse et finalement tombe hors du champ de vision de la caméra, puis pressez la touche *[Echap]* pour mettre fin au jeu. Immédiatement après la fin de la simulation, l'éditeur de courbes IPO se met automatiquement à jour pour révéler les courbes IPO enregistrées, pour l'objet balle, au cours de la simulation.



Il apparaît vite comme évident que la mise en place de telles courbes, manuellement, dans le cadre d'une animation conventionnelle, aurait été particulièrement délicate.

Les images qui suivent présentent quelques moments choisis de l'animation finalement obtenue, sans trop d'efforts de notre part ! Certes, la démarche consistant à faire appel au moteur de jeu pour animer une balle tombant dans notre scène n'est pas forcément la plus logique ; elle n'est possible que depuis récemment dans Blender, et il n'est pas exclu, dans le futur, qu'une animation conventionnelle, qui tient compte des lois de la physique, puisse être menée directement.

**Figure 8-30**

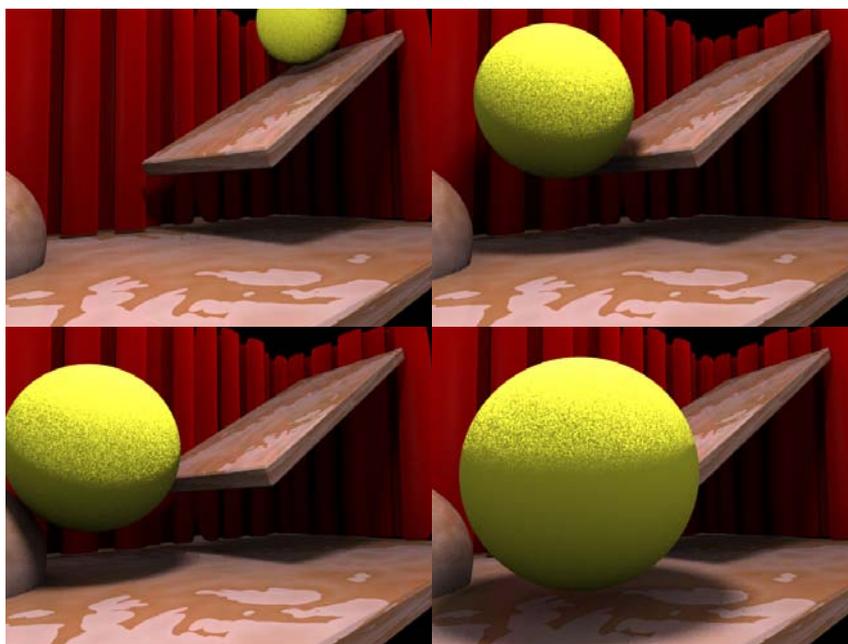
Les courbes IPO de la sphère, telles que construites par le simulateur physique

#### MÉTHODE Animation trop longue à jouer !

Cela peut arriver très rapidement si vous utilisez des modèles présentant un trop gros nombre de points de contrôle. Pour jouer l'animation (touche *[P]*) le plus vite possible :

- Préférez le mode fil de fer au mode ombré.
- Mettez les objets de la scène n'intervenant pas dans l'animation (en particulier, ceux n'entrant jamais en collision avec les objets animés) sur un calque différent, que vous n'affichez pas.
- Choisissez la forme de la frontière *Bounds* pour qu'elle corresponde au mieux à celle de l'objet animé, tout en étant la plus simple possible.

Si vous avez besoin de contrôler la qualité des collisions, par exemple, vérifiez votre animation en vue de caméra en mode ombré en appuyant sur la combinaison de touches *[Alt]+[A]*.



**Figure 8-31**  
Quelques images de l'animation  
résultante de corps rigides

## La simulation de fluides

Un autre domaine d'animation pour lequel Blender est doté d'un système de simulation avancé est celui des fluides. Son usage n'est guère plus compliqué que les autres, mais fait appel à des notions sensiblement différentes, en particulier celle de « domaine ».

Du point de vue du simulateur, trois types d'objets sont pris en considération :

- le domaine de calcul ;
- les obstacles ;
- les objets fluides.

Le domaine de calcul permet de circonscrire une zone physique pour laquelle les calculs seront effectués. C'est par ailleurs dans les paramètres du domaine que seront définies les propriétés des fluides simulés : la résolution du fluide, sa viscosité, et la durée de la simulation.

Les obstacles sont des objets situés dans le domaine. Le système détectera les éventuelles collisions entre les fluides en mouvement et les obstacles, de sorte que les fluides pourront les contourner ou ruisseler sur eux.

Les objets fluides peuvent être de nature plus variée, et correspondre à des volumes de liquide existants, à une source de liquide, ou enfin à une évacuation de liquide.