

Rapport Soutenance

Block’N’Roll

Soutenance 1

Loïc Bellonnet-Mottet (Sup B1)

Arthur Ulpas (Sup C1)

Alexandre Yvart (Sup A1)

Louis Dufeu (Sup B1)

Suspicious Lama Inc.

Mardi 7 Mars 2017

Sommaire

1	Introduction	3
2	Avancement par tâches	4
2.1	Déplacements du joueur	4
2.2	Mécaniques de jeu	5
2.3	Création de niveaux	7
2.4	Multijoueur	8
2.5	Graphismes	8
2.6	Sons	8
2.7	Site Internet	8
3	Prévisions pour la prochaine soutenance	9
3.1	Déplacements du joueur	9
3.2	Mécaniques de jeu	10
3.3	Création de niveaux	11
3.4	Multijoueur	11
3.5	Graphismes	11
3.6	Sons	11
3.7	Site Internet	11
4	Les points positifs jusqu'à maintenant	12
5	Les points négatifs jusqu'à maintenant	13
6	Conclusion	14

1 Introduction

Bienvenue dans ce premier rapport de soutenance qui va vous permettre d'avoir un aperçu des différentes avancées du projet depuis la remise du cahier des charges.

Pour résumer en quelques phrase l'idée du projet, Block'N'Roll est un jeu vidéo de puzzle et de plateformes se déroulant dans un monde cubique. Le joueur contrôle un cube qui se déplace en roulant. Je vous sens perplexe, mais si, un cube peut tout à fait rouler, ce n'est pas un abus de langage : <https://en.wikipedia.org/wiki/Roll>. Le joueur évolue donc dans les niveaux composés de puzzles et autres obstacles afin de récupérer des objectifs pour activer la porte de sortie et progresser vers le niveau suivant.

Sans plus attendre, voici les éléments déjà présents dans le jeu, ceux que nous comptons ajouter dans le futur et les points positifs et négatifs auxquels nous avons pu nous heurter durant les débuts de la réalisation du projet.



2 Avancement par tâches

2.1 Déplacements du joueur

Responsable: Louis / Suppléant: Loïc / Prévion pour la soutenance 1: 70%

Nous avons prévu un avancement de 70% par rapport aux déplacements du cube sur le cahiers des charges et je pense que nous l'avons effectué pour la première soutenance. Voici le résumé de ce qui a été implémentés pour les déplacements du joueur.

Marche et Sprint: (Louis + Loïc)

Bien sûr, le cube peut se déplacer dans les 4 directions en utilisant les touches ZQSD. Le cube ne se déplace pas en glissant sur le sol, mais plutôt en roulant sur l'une de ses arêtes en fonction de la touche appuyée comme sur le logo du jeu. Le cube ne se déplace que d'une unité à la fois puisque nous considérons le monde comme une grille en 3 dimensions. Ainsi, tous les murs et ennemis du jeu se rapprochent, plus ou moins, de la forme d'un cube ou d'un pavé. En maintenant la touche Shift, le cube effectue ce même déplacement mais plus vite, comme un sprint classique.

Saut: (Louis + Loïc)

En plus de se déplacer au sol, le cube peut sauter en appuyant sur la barre d'espace. En l'air, le cube peut se décaler d'une case sur le côté pour monter un mur d'une hauteur d'une unité maximum.

Grimper: (Louis)

Même si sauter est un bon moyen de gagner de la hauteur, ce n'est pas forcément très esthétique (nous ne sommes d'ailleurs pas sûr de conserver le saut). C'est pourquoi le cube peut grimper sur un mur. En effet, lorsque le cube est directement adjacent à un mur, le joueur peut se déplacer en direction du mur en question et peut commencer à "rouler" contre le mur. Comme pour la marche et le sprint, le cube fait une rotation autour d'une arête du mur. A ce moment-là, le joueur peut maintenir la touche pour progressivement atteindre le sommet du mur. Si le joueur relâche la touche, le cube redescend progressivement à sa position de départ. Avec de l'entraînement et un timing précis, le joueur peut se maintenir dans cette position, suspendu en l'air. Cette mécanique de jeu pourra éventuellement s'avérer utile dans le futur.

Gestion de la Caméra: (Louis)

Cela peut sembler étrange mais la caméra est un élément à considérer pendant le déplacement du joueur. En effet, le joueur déplace un cube, c'est pourquoi quand le cube roule sur le sol, la hauteur du centre du cube n'est pas constante. Cela se ressent sur la caméra avec un petit sauttillement à chaque déplacement, ce qui peut rendre nauséeux rapidement. Ainsi, nous avons fait en sorte que la caméra reste à la même hauteur tout en suivant le cube. De plus, la caméra peut pivoter autour du joueur pour avoir un autre aperçu du niveau en cours avec les touches fléchées Gauche et Droite. Comme la caméra change d'orientation suite à la rotation, les touches pour déplacer le cube sont modifiées pour que le joueur ne se perde pas. Par ailleurs, le joueur peut également dézoomer la caméra en appuyant sur la touche fléchée Bas.

2.2 Mécaniques de jeu

Responsable: Loïc / Suppléant: Arthur / Prévision pour la soutenance 1: 25%

Il est difficile de savoir si la prédiction de l'avancement de la tâche a été respecté car les idées pour les différentes mécaniques viennent au fur et à mesure que l'on avance dans le projet. Malgré cela, je pense que les 25% prévus pour cette soutenance sont atteints, voire légèrement dépassés.

Voici la liste et le fonctionnement des différents mécaniques de jeu implémentées depuis la remise du cahier des charges:

Les portes et boutons: (Loïc)

Les portes sont en apparence (pour l'instant) identiques en tout points à des murs. La seule chose qui les différencie est le fait qu'une porte peut être associée à un bouton (représenté par une petite dalle au sol). Lorsque le joueur passe sur ce bouton, la porte s'abaisse. Ces deux éléments sont faits de telle manière que rien n'est à modifier si la taille / forme de la porte est modifiée. Cela est directement pris en charge par un script.

Les objectifs et la sortie du niveau: (Loïc)

Les objectifs sont représentés par des cubes qui flottent en l'air en tournant, que le joueur doit récupérer pour activer la porte de sortie et continuer vers un autre niveau. Un script modifie la couleur des objectifs en fonction du temps, de telle sorte à ce qu'ils aient une couleur "arc-en-ciel". La sortie, représentée par une dalle similaire à celle des boutons, est inactive tant qu'il reste des objectifs à récupérer dans le niveau. Une fois tous les objectifs ramassés, elle s'illumine en émettant des particules, indiquant au joueur qu'il peut accéder au niveau suivant.

Les ennemis: (Loïc)

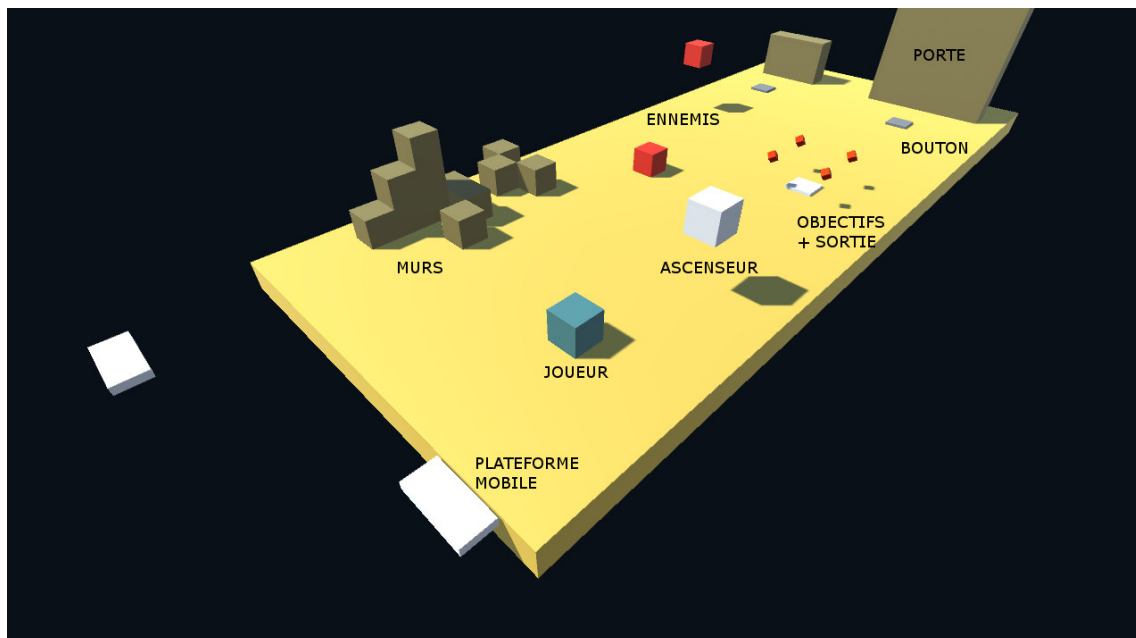
Les ennemis sont représentés par des cubes rouges, d'une taille légèrement inférieur à celle du joueur. Ils se déplacent en glissant à une vitesse variable sur le sol en suivant un chemin déterminé par un certain nombre de points. On peut très facilement modifier son parcours en ajoutant ou en retirant des objets représentant les différents points de passage dans un tableau géré par un script. Lorsque qu'une collision entre le joueur et un ennemi se produit, le cube du joueur se décompose en 64 plus petits cubes qui tombent et roulent au sol. Le niveau est ensuite réinitialisé au bout de deux secondes.

Les plateformes mobiles: (Loïc)

Les plateformes sont représentées par des dalles un peu plus épaisses que les boutons ou la sortie, sauf qu'elle flottent en l'air. Il y a différentes options pour les plateformes que l'on peut gérer directement depuis l'inspecteur unity sans avoir à repasser par le script. Le système de déplacement des plateformes est le même que celui des ennemis, elles alternent entre un point A et un point B (puis C, D etc ... au besoin) à une vitesse variable. Le premier type de plateformes sont celles qui font des allers retours entre les différents points du chemin sans s'arrêter, il faut donc faire preuve d'adresse pour pouvoir monter ou descendre de ces plateformes. Il y a ensuite les plateformes qui effectuent une seule fois le chemin défini par les différents point, et enfin celles qui ne commencent à bouger qu'une fois que le joueur est monté dessus. On peut donc obtenir des effets différents en sélectionnant une ou plusieurs de ces options.

Les ascenseurs: (Arthur)

Les ascenseurs sont des cubes blancs qui montent et redescendent à un intervalle de temps que l'on peut directement choisir depuis l'inspecteur Unity. L'ascenseur passe à travers les ennemis mais il ne passe pas à travers le joueur, ce qui peut permettre différentes combinaisons, on peut par exemple mettre un ascenseur au milieu du « chemin » d'un ennemi et le joueur devrait attendre le bon moment pour passer. Les ascenseurs ne fonctionnent pas exactement comme les plateformes, en effet, ces derniers ne se déplacent qu'horizontalement et ne se déplacent pas en permanence à la même vitesse, ils ralentissent avant d'arriver à leur position. Cette vitesse peut être réglée depuis l'inspecteur Unity.



2.3 Création de niveaux

Responsable: Arthur / Suppléant: Louis / Prévision pour la soutenance 1: 20%

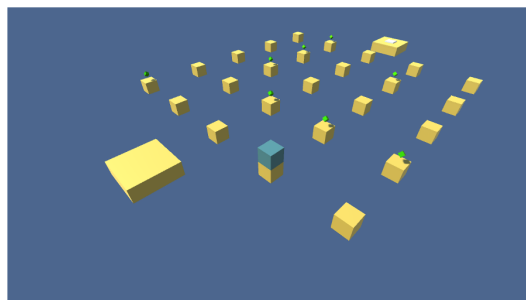
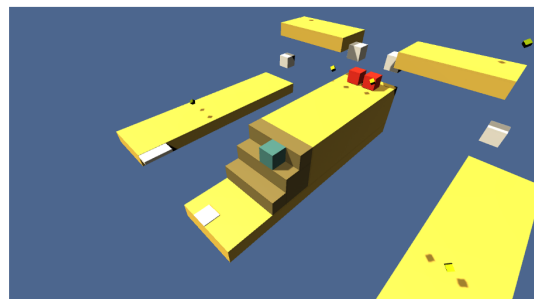
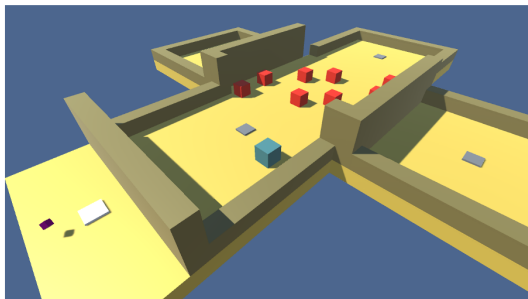
Les niveaux qui ont été créés jusqu'ici essaient de montrer ce que l'on peut faire avec les mécaniques existantes.

Le premier niveau exploite les portes, le joueur a besoin de se déplacer en évitant les ennemis pour pouvoir ouvrir les portes et accéder à la fin du niveau.

Le deuxième niveau, quant à lui, consiste à trouver le chemin permettant d'accéder à la fin en récupérant les objectifs sans tomber. Le joueur doit essayer plusieurs fois avant de trouver le bon chemin (die and retry).

Le troisième niveau utilise les ascenseurs et les escaliers de sorte à utiliser la verticalité du jeu. Le joueur doit esquiver les ennemis pour accéder à un ascenseur et récupérer les objectifs.

Nous avons également créé un niveau pour la soutenance qui a pour but de montrer toutes nos mécaniques de jeu.



2.4 Multijoueur

Responsable: Alexandre / Suppléant: Loïc / Prévision pour la soutenance 1: 5%

La gestion du multijoueur a nécessité quelques recherches internet, étant donné que je n'avais strictement aucune notion sur la gestion de serveur etc. En effet, il existe certains outils fournis par Unity afin de gérer les fonctions en ligne, comme par exemple la possibilité d'utiliser leurs serveurs pour se connecter au matchmaking. Cela n'était pas intéressant pour commencer, étant donné qu'il fallait déjà que le multijoueur en LAN marche avant de tenter de le relier à des serveurs.

La difficulté était de faire fonctionner le multijoueur sans influencer sur le jeu solo, mais aussi de savoir différencier les joueurs (ce qui a été fait graphiquement en changeant la couleur du joueur contrôlé par exemple), c'est-à-dire ne pouvoir agir que sur le cube qui nous appartient. La connexion en multijoueur sur le même ordinateur a été assez simple à gérer, c'était donc la partie la plus rapide du travail. Les 5% de progression sur le multijoueur ont probablement été dépassés, cependant le plus dur reste à faire.

2.5 Graphismes

Responsable: Alexandre / Suppléant: Loïc / Prévision pour la soutenance 1: 10%

N'ayant pas besoin de graphismes poussés pour notre jeu, les 10% ont probablement été atteints avec uniquement le logo du jeu créé par Alexandre. Ne pensant pas créer de textures personnalisées pour les niveaux du jeu, le reste de la partie graphisme portera sur le site internet et le menu.

2.6 Sons

Responsable: Louis / Suppléant: Arthur / Prévision pour la soutenance 1: 10%

Les 10% de progression annoncés étaient peut-être un petit peu élevés car nous approchons plus des ... 0%. Les sons n'étant pas une partie majeure du jeu, nous nous sommes concentrés sur les autres aspects de celui-ci, oubliant complètement les sons. Nous nous rattrapons pour la soutenance suivante !

2.7 Site Internet

Responsable: Arthur / Suppléant: Alexandre / Prévision pour la soutenance 1: 0%

Le site internet n'étant pas obligatoire pour la première soutenance, et n'aidant en rien à l'avancée du jeu, la prévision de 0% a été totalement respectée.

3 Prévisions pour la prochaine soutenance

3.1 Déplacements du joueur

Se maintenir sur des plateformes qui se déplacent

Comme dit précédemment, le cube peut progressivement grimper sur un mur voire se maintenir en l'air avec un bon timing. La prochaine étape serait que le cube puisse rester en l'air pendant qu'une plateforme se déplace, pour franchir un gouffre ou atteindre des zones autrement inaccessibles par exemple.

Changement de Gravité :

Pour la prochaine soutenance, nous avons pensé à implémenter un bouton ou un trigger pour inverser la gravité. En conséquence de cela, le cube tomberait vers le haut, sauterait vers le bas par exemple. Cela permettrait de mettre en place des énigmes différentes et variées comparées à ce que nous avons déjà pensé. De plus, nous pourrions non seulement inverser le sens de la gravité mais aussi la direction, pour se déplacer sur les murs par exemple. Bien sûr, lors de ces changements gravitationnels, il faudra prendre en compte et modifier les contrôles pour déplacer le cube et la même chose pour la caméra.

Changement de Forme :

Dans le cahier des charges, nous avons formulé la possibilité de changer de forme, passer de cube à sphère par exemple. Il est parfaitement faisable de mettre en place cette capacité pour la prochaine soutenance. Les mouvements du joueur seront ainsi modifiés, de même pour les contrôles et la caméra. Le faire pour la prochaine soutenance permettrait de commencer l'assemblage des niveaux du jeu.

Blink :

Il s'agit d'une capacité du joueur permettant de se téléporter dans une certaine direction et avec une portée limitée. Par exemple, en appuyant sur la touche B puis sur une touche de direction Z/Q/S/D/Space, le cube se téléportera dans la direction souhaitée. De plus, pour plus de clarté, en maintenant la touche de direction au lieu d'appuyer, le joueur pourrait avoir une prévisualisation de l'endroit où il arrivera et si la téléportation est possible et nous pourrions instaurer un cooldown pour empêcher le joueur d'abuser de cette capacité trop souvent.

Limitations :

Même si ajouter des capacités est très intéressant, pour certains niveaux, nous n'avons pas forcément envie que le joueur ait accès au changement de gravité et à la téléportation. Ainsi, certains niveaux pourraient modifier le comportement du joueur : imposer une forme au joueur, réduire ou augmenter la portée de la téléportation suite à l'activation d'un interrupteur, faire un changement de gravité toutes les 5 secondes dans une direction différente par exemple. Même si ces changements sont spécifiques au niveau en cours, ils se rapportent tous au mouvement du joueur et de la caméra.

3.2 Mécaniques de jeu

Les mécaniques de jeu étant "le cœur" d'un jeu vidéo, il faudra continuer d'en ajouter pour les soutenances suivantes. Heureusement, les idées ne manquent pas encore !

Voici ce que nous comptons implémenter pour la prochaine soutenance:

Des plateformes à passage limités:

Des plateformes fixes qui tombent une fois que le joueur est passé dessus. On pourrait envisager un nombre de passages variable, indiqué par la couleur de la plateforme qui changerait quand le joueur passe dessus (vert = 3 passages, jaune = 2 passages, orange = 1 etc ...).

Un système de dalles "on / off":

Des dalles à l'état actif ou inactif qui changeraient d'état à chaque passage du joueur. Il faudrait que toutes les dalles soient dans l'état actif pour débloquer une porte / un objectif / activer la sortie ou autres ...

Un système de téléportation:

Deux dalles reliées par un script qui permettraient au joueur de se téléporter de l'une à autre (sous certaines conditions ou non).

De nouveaux ennemis:

Des ennemis qui tirent des projectiles par exemple.

Améliorations:

Apporter des améliorations sur les ascenseurs et plateformes mobiles (y ajouter certaines options).

3.3 Création de niveaux

Pour la prochaine soutenance, nous espérons avoir environ une dizaine de niveaux pour notre jeu. Le but serait de montrer toutes les possibilités de nos différentes mécaniques de jeu, d'autant plus que nous aurons sûrement de nouvelles mécaniques nous permettant encore plus de possibilités. Pour créer ces niveaux, nous allons nous inspirer d'autres puzzle games / jeux de réflexion tels que Edge ou encore Portal. En effet, il est important pour nous de voir comment sont faits les niveaux de ces jeux car ils n'ont pas forcément beaucoup de mécaniques de jeu différentes mais arrivent tout de même à proposer des niveaux tous très différents.

3.4 Multijoueur

Faire que la caméra suive le cube que l'on contrôle, offrir la possibilité de jouer en multijoueur à partir de deux ordinateurs différents, adapter éventuellement certaines mécaniques de gameplay au multijoueur (comme par exemple le saut), en enfin créer des niveaux pour pouvoir exploiter le potentiel du jeu à plusieurs.

3.5 Graphismes

Créer un fond pour les différents niveaux (peut être un fond noir étoilé pour donner un côté spatial ?). Commencer à créer un menu pour le jeu et les différentes images / vidéos nécessaires pour le site internet.

3.6 Sons

Trouver des sons chiptune pour les déplacements du joueur et la destruction du joueur (collision avec un ennemi) au minimum, puis éventuellement pour les portes et le ramassage d'un objectif par le joueur et les implémenter dans le jeu.

3.7 Site Internet

Le site internet aura pour but de présenter notre groupe de projet, renseigner les gens sur notre jeu en montrant des vidéos de gameplay et des screenshots ainsi que quelques explications sur le jeu. Pour la prochaine soutenance, nous ne présenterons qu'une version locale du site, nous l'hébergerons pour la troisième soutenance.

4 Les points positifs jusqu'à maintenant

Tous les membres sont d'accord pour dire que l'ambiance du groupe est agréable, tout comme de travailler sur ce projet.

La réalisation de ce projet est très enrichissante au niveau de la programmation et nous aide à trouver des solutions à nos problèmes par nous même. Comme la plupart du temps les réponses que l'on peut trouver à certaines questions sur internet ne correspondent pas exactement à nos problèmes, nous devons adapter cette solution ou s'en inspirer afin de créer notre propre solution. La communauté de personnes utilisant unity dans les forums a l'air très amicale et sympathique.

Encore une fois, nous voulons insister sur ce point: il est très agréable de pouvoir enfin se servir de nos connaissances en mathématiques acquises depuis le collège dans un projet concret, chose que nous n'avions jamais eu l'occasion de faire (à moins que vous considériez que calculer une force ou une intégrale en cours est quelque chose concret ...). En effet, notre jeu étant constitué de cubes, tout est symétrique et des règles assez strictes doivent être appliquées. Nous avons par exemple utilisé des formules de trigonométrie pour déterminer les mouvements de rotation et de translation du cube.

De plus, l'utilisation du Unity Cloud rend le travail de groupe et le partage des fichiers très plaisant car nous pouvons tous travailler et envoyer les modifications effectuées presque en même temps afin que tout le monde travaille avec les dernière versions du jeu.

5 Les points négatifs jusqu'à maintenant

Le plus gros problème que nous ayons eu dans la réalisation de ce projet jusqu'à maintenant est l'utilisation de GitHub pour partager le projet entre les membres. Aucun membre du groupe n'ayant utilisé Git jusqu'à présent, les débuts de l'utilisation de GitHub furent assez laborieux.

Une fois le repository pour le projet mis en place, nous avons pu commencer à travailler en prenant en compte les avancements des autres. Tout se passait merveilleusement bien tant que nous travaillions chacun notre tour sur le projet.

Les problèmes commencèrent lorsque nous avons décidé de travailler EN MEME TEMPS. Le commit de la première personne ayant d'arrêté de travailler s'est bien passé. Mais lorsque le second décida également de commit ses changements, tout est devenu compliqué. Les SPE nous ayant dit que "c'est facile, vous avez juste à faire un git merge !", c'est-ce que nous avons essayé de faire. Malheureusement Git ne peut apparemment pas fusionner les scènes Unity, ce qui, vous en conviendrez, n'est pas très pratique quand on est censé faire un jeu sous Unity.

Nous avons donc trouvé un plugin permettant à Git de fusionner les scènes et assets Unity sous la supervision d'un membre du projet. Mais il n'était disponible uniquement sous Unity 4... L'idée d'utiliser Git a alors été abandonnée et nous sommes passés sur le Unity cloud. Malgré le fait qu'il nous ait été déconseillé par les SPE (sans raison apparente), son utilisation est tout de même beaucoup plus simple ! Les fichiers partagés sont très légers, les scènes et scripts modifiés sont fusionnés sans aucun soucis etc ...

Nous ne sommes pas en train de dire que Git c'est mal, que ce n'est pas un bon logiciel ou quoi que se soit. Il doit être absolument formidable quand il s'agit de partager du code uniquement. Mais pour l'utilisation avec Unity, quel calvaire ...

6 Conclusion

Dans l'ensemble (sauf pour les sons) le planning a été respecté et le résultat obtenu est, selon nous, tout à fait honorable. Nous avons maintenant une base solide qu'il faudra continuer d'améliorer pour les soutenances suivantes. Mais maintenant que nous possédons les éléments de base de notre jeu, il va falloir ajouter les détails (sons, graphismes, menu etc ...) néanmoins essentiels dans la réalisation d'un jeu cohérent.

Notre vision du projet est pour l'instant très positive grâce à une bonne cohésion de groupe et nous espérons que cela pourra continuer dans la même direction.