

MR4

Audio global design

1-Les musiques

2-Le sound design

3-Nos Besoins

4-Interogations

5-Les contacts

1-Les musiques

Mode solo :

Un circuit = un background bass/drums/accord  
Un personnage = phrases musicales typées en fonction du character design

*Exemple :  
Background\_Circuit\_01 : style musique 01   
Perso\_01 : lead type mélodie ou solo guitare électrique   
-changement de personnage -  
Background\_Circuit\_01 : style musique 01  
Perso\_02 : lead type mélodie ou solo orgue*

Mode multi :

Un circuit = un background bass/drums/accord  
Un personnage = phrases musicales typées en fonction du character design

*Exemple en écoute sur le ftp dans musics/proto :* ***MR\_Mus\_Montage.wav***

*En début de course le tempo est progressif, il accélère jusqu’à 200 km/h, à partir de là commence une bataille de solo en fonction du joueur qui est devant, il faut écouter les instruments en jaune.*

*Sur le ftp dans musics/proto/integration/* *Multitrack\_01 :*

*Les fichiers correspondant au tableur ci-dessous permettant de faire des tests d’intégration sur le mode multi.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **T.C.** | **Type de trigger** | **Description gameplay** | **Instrument** | **Instructions** |
| 00.18 | Vitesse | Le motard passe le cap des 200 km/h | Batterie | Démuter la piste « FastSpeed » en fade in |
| 00.27 | Personnage | *Perso\_01*tête du battle | Guitare Solo |  |
| 00.39 | Personnage | *Perso\_02* double  *Perso\_01* | Orgue Solo | Muter la piste « Lead1 » en fade out et démuter la piste « Lead2 » en fade in |
| 00.45 | Personnage | *Perso\_01* double  *Perso\_02* | Guitare Solo | Muter la piste « Lead2 » en fade out et démuter la piste « Lead1 » en fade in |
| 00.49 | Personnage | *Perso\_02* double  *Perso\_01* | Orgue Solo | Muter la piste « Lead1 » en fade out et démuter la piste « Lead2 » en fade in |

2-Sound Design

Sources sonores In Game :

1) La moto (moteur, crissement de pneu, collision)

2) Les éléments du décors collisionable (expl :plots)

3) Les éléments du décor non collisionnable (expl : public, court d'eau)

4) Les éléments de HUD (feedbacks)

1) Le son de la moto

Afin de rendre le comportement de la moto le plus réaliste possible, nous avons besoin de prise de son de moteur d'une même moto.

• Idle de stop (le moteur tourne au point mort, loop de 20 sec)

• Idle à différentes vitesses/allure *(loop de 20-30 sec de moteur tournant à une vitesse donnée, de la façon la plus stable possible, sans freinage ni accélération)* une dizaine serait confortable, de la vitesse la plus basse à la plus haute

• Plusieurs accélérations constantes de différentes vitesses du stop au max

• Plusieurs sons de passage de vitesse (décrochage)

• Sons de freinage/pneus

• Sons de démarrage

Comportement du moteur :

Le son du moteur sera relié à l'accélération et la vitesse du véhicule via un RTPC, et sera constitué d'une succession de Idle de moto couplé à une utilisation du pitch shift.

**La question est de savoir si dans Moto Racer les passages de vitesse se font automatiquement ou manuellement, ou les deux.** Cette information est capitale pour pouvoir prototyper de façon optimale.

2) Les éléments du décor collisionnables

Il nous sera nécessaire de connaître à l'avance les objets avec lequel la moto est susceptible d'entrer en collision (typiquement des plots). Il nous sera ensuite plus facile d'anticiper sur les sons à appliquer sur le mouvement de ces objets.

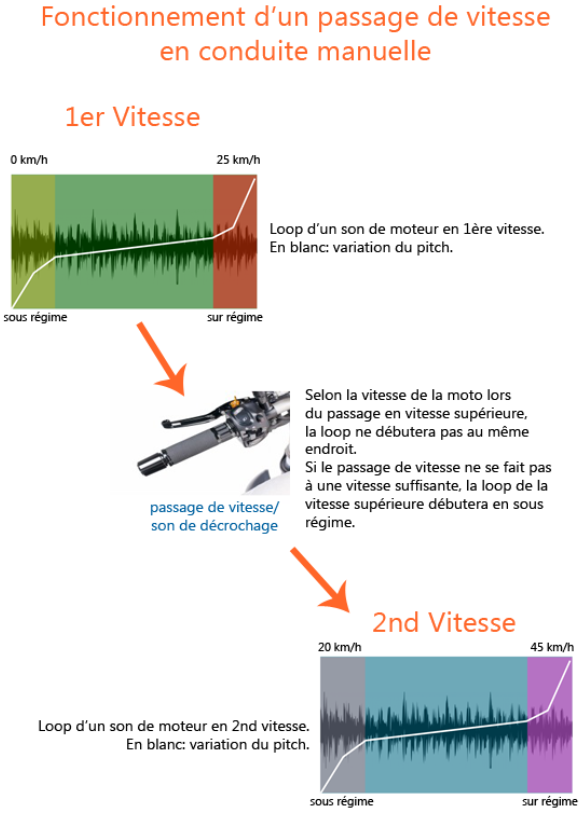
3) Les éléments du décor non collisionnables

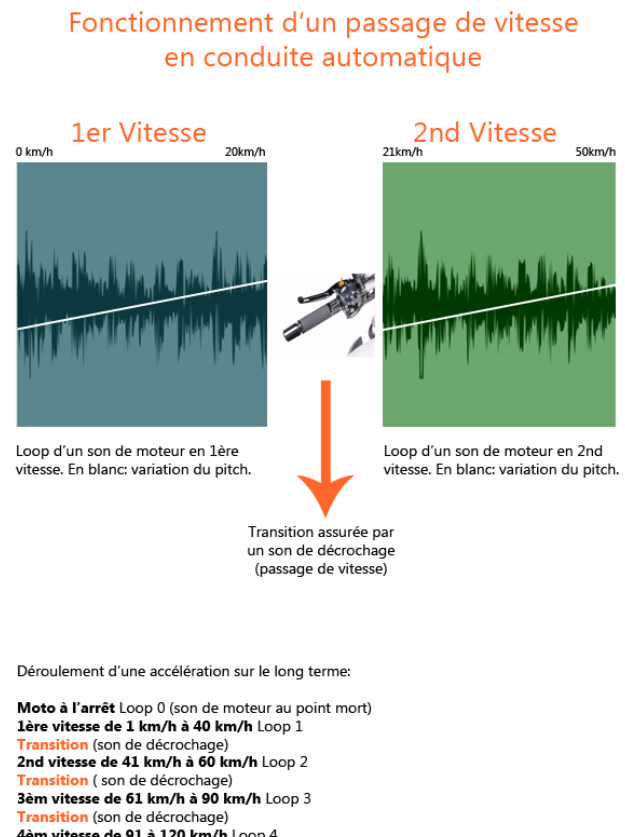
Comme pour les éléments collisionnables, il nous sera nécessaire de connaître les tracés des circuits et les éléments du décor susceptibles d'être entendu par le joueur à leur proximité (par exemple, un court d'eau, une tribune de spectateur). UDK possède une fonctionnalité de placement d'objet sonore avec courbe d'atténuation paramétrable qui simplifiera le travail de placement de ces sons.Pour un plus grand réalisme et sensation de vitesse, les objets du décor (*barrière de sécu, arbres, bâtiments,…*) à proximité du joueur provoqueront le lancement d'un son type woosh (air) couplé à une modulation en fréquence (pitch*)*.

4) Les éléments de HUD

Il nous faudra une liste des informations et feedbacks nécessaires au joueur durant une partie (info sur les nombres de tours).

*Les sons de l’interface utilisateur seront du type moderne (voir exemple) pour les boutons, et des sons réalistes pour les transitions de menu, une onomatopée sera déclenchée au changement de personnage, un son de moto sera déclenché au changement de véhicule. (proposition)*





3-Nos besoins pour estimation de la quantité de travail :

* Le nombre de musiques
* Une estimation des sfxs de votre côté

|  |
| --- |
| Menu |
| Button |
| UI/GUI |
| Jauge / compteur |
| Surfaces |
| Motos |
| Anim personnages |
| FMV/RTC |
| Anim decors |
| Objet pickup |
| SplashScreen |

* Visuels des décors
* Visuels personnages
* Visuels motos
* Visuels HUD
* Visuels Menu

4-Questionements Divers :

* Mise en place d’un side chaine pour un mix optimal
* Vitesse maximale d’une moto avec découpe par tranche de vitesses
* Control du rendu/mix de l’intégration du son par APS
* Les nomenclatures
* Son réel de moto ou synthèse temps réel

Contacts :

Lucas Lead Sound designer: [l.rousselot@audioplumstudio.com](mailto:l.rousselot@audioplumstudio.com)

Sacha Sound designer : [s.holsnyder@audioplumstudio.com](mailto:s.holsnyder@audioplumstudio.com)

Laurent Compositeur : [l.peju@audioplumstudio.com](mailto:l.peju@audioplumstudio.com)

Alexandre Chef de projet Audio : [a.baranger@audioplumstudio.com](mailto:a.baranger@audioplumstudio.com)

Compte FTP dossier MR4 :

Adresse : APS.myftp.biz

User : ARTEFACTS2015M

Passe : MAPS0101

Port : 2121