

1	Présentation de jeu	2
1.1	Star Citizen	2
1.2	Sortie fichier.....	3
1.4	Expression de besoin : Le profilage	3
2	Le Logiciel Joytokey.....	3
2.1	Le logiciel	3
2.2	Sortie fichier du logiciel	3
2.3	Souci de synthèse (visualisation)	3
3	Résumé des objectifs.....	4
3.1	Synthèse du jeu avec Joytokey	4
3.2	Adaptation en fonction du développement (jeu en alpha)	4
3.2.1.1	Exportation de la configuration basique.....	4
3.2.1.2	Création d'un fichier etoile.xml	4
3.3	Mise en ligne pour utilisateur.....	5
4	Organigrammes	7
4.1	Organigramme niveau 1	7
4.2	Organigramme Niveau 2: Adaptabilité au développement du jeu.....	8
4.3	Organigramme Niveau 3 : Mise en ligne pour interface utilisateur	10

Figure 1	Hotas choisi.....	2
Figure 2	Image de combat spatial (Dogfight)	Erreur ! Signet non défini.
Figure 3	visualisation du fichier par défaut	4
Figure 4	Nombre de fonctions dans etoile.xml.....	5
Figure 5	Nombre de catégories dans etoile.xml	5

1 Présentation de jeu

1.1 Star Citizen

Son développement est en cours (Alpha test).

Dans celui-ci, le joueur incarnera dans un 'monde ouvert' (GTA-like), un citoyen des étoiles. Tout se passera dans le futur dans un contexte de science-fiction (Space Opera).

<https://robertsspaceindustries.com/>

Le joueur aura l'occasion de jouer en se déplaçant à pied, mais aussi en pilotant des vaisseaux. Pour ce dernier mode de déplacement, le jeu prévoit non seulement l'utilisation du clavier/souris, mais aussi le joystick (HOTAS).



Figure 1 Hotas choisi

Pour ma part, en terme d'immersion, je préfère l'utilisation du Hotas. J'ai choisi le T Flight (Thrustmaster). Il se compose de 12 boutons numériques, et de 5 axes analogiques.



Figure 2 Image de combat spatial (Dogfight)

1.2 Sortie fichier

Une fois l'extraction de l'affectation des touches (keybinding) faite dans le jeu, un fichier (xml) apparait dans le répertoire du jeu.

1.3 limite du keybinding

1.4 Expression de besoin : Le profilage

Partisan donc de l'utilisation du hotas dans le jeu, les « aller/retour » du joystick par rapport au clavier, limite la jouabilité. L'objectif reste de 'multiplier' les boutons du hotas en créant des profils 'remapping', en fonction de la configuration du pilotage du vaisseau (pilotage simple, attaque, défense, métier spécifique).

2 Le Logiciel Joytokey

2.1 Le logiciel

2.2 Sortie fichier du logiciel

Chaque profil génère un fichier texte (.cfg).

2.3 Souci de synthèse (visualisation)

Les sous profils ne sont pas visualisés en temps réel.

3 Résumé des objectifs

3.1 Synthèse du jeu avec Joytokey

(Organigramme niv.1)

Créer une interface d'un jeu (Star Citizen), qui synthétisera et visualisera (sur un deuxième écran) les besoins de ce dernier en terme de keybinding (affectation des touches du clavier), avec l'aide d'un logiciel (Joytokey) de remapping (profilage) d'un HOTAS (joystick).

3.2 Adaptation en fonction du développement (jeu en alpha)

3.2.1 Recherche des fonctionnalités et des catégories du keybinding

(Organigramme niv.2)

Le jeu étant encore au stade alpha de développement, de nouveaux métiers vont survenir. Effectivement on a vu apparaître le métier de mineur de minéraux lors de la dernière mise à jour (alpha3.2). D'autres sont à venir : chasseur de prime, ambulancier... A chaque nouvelle implémentation, le keybinding se voit alors se doter d'autant de nouvelles fonctions.

3.2.1.1 Exportation de la configuration basique

Il existe cependant un problème lors de la recherche de toutes les fonctionnalités du jeu. En effet, lors de l'extraction de la configuration de base, le fichier alors créé n'en comporte aucune (annexe1). Pour reformuler le problème, seules les fonctionnalités modifiées de la configuration de base apparaissent.

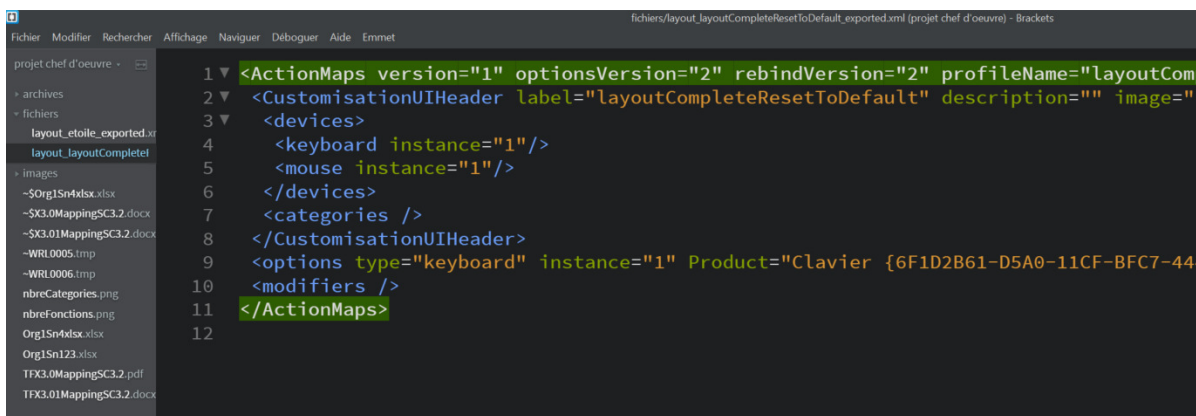


Figure 3 visualisation du fichier par défaut

3.2.1.2 Création d'un fichier etoile.xml

Pour contourner ce problème, il s'agit de réaffecter toutes les fonctions par une valeur différente de celle qui lui est attribuer par défaut. Ainsi, toutes celles-là apparaîtront lors de l'extraction.

On choisira arbitrairement l'attribution '*' pour toutes les fonctions.

On trouvera donc 216 fonctions à traiter...

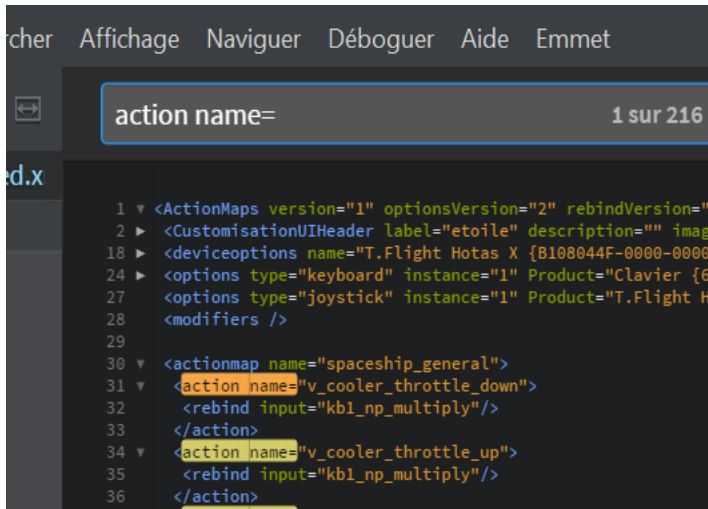
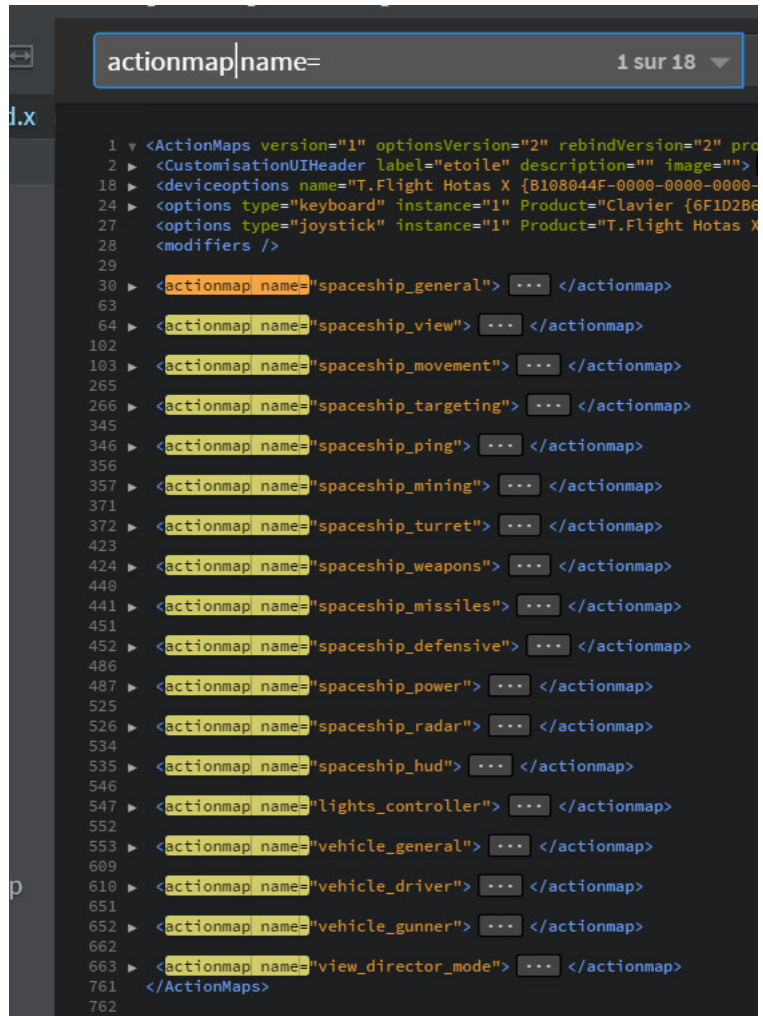


Figure 4 Nombre de fonctions dans etoile.xml

...Réparti en 18 catégories:



<C:\Users\fef\AppData\Roaming\Microsoft\Word\images\FichierEtoile.png>

3.2.2

3.3 Mise en ligne pour utilisateur

(Organigramme niv.3)

Figure 5 Nombre de catégories dans etoile.xml

3.4 Roadmap

1/ importationDefault.php

On crée un fichier 'layout_etoile_exported.xml' (toutes les fonctions sont affectées par la valeur * du pavé numérique).

On crée un fichier resetToDefault.xml (affectation de base du jeu)

Par importDéfaut.php :

On importe 'layout_etoile_exported.xml' via sur variableParDefault.sql.

On écrase ce dernier.avec le contenu de resetToDefault.xml

2/ importationSc.php

On importe variablesParDefaut.sql.

On écrase celui-ci par la configuration utilisateur (layout_user_exported.xml)

3-4-5-6/ Importation des fichiers cfg de Joytokey

7/ syntheseKeybindingProfils.php

On importe syntheseSc.sql, et on l'écrase par syntheseJ2KProfil1.sql, syntheseJ2KProfil2.sql, syntheseJ2KProfil3.sql, syntheseJ2KProfil4.sql.

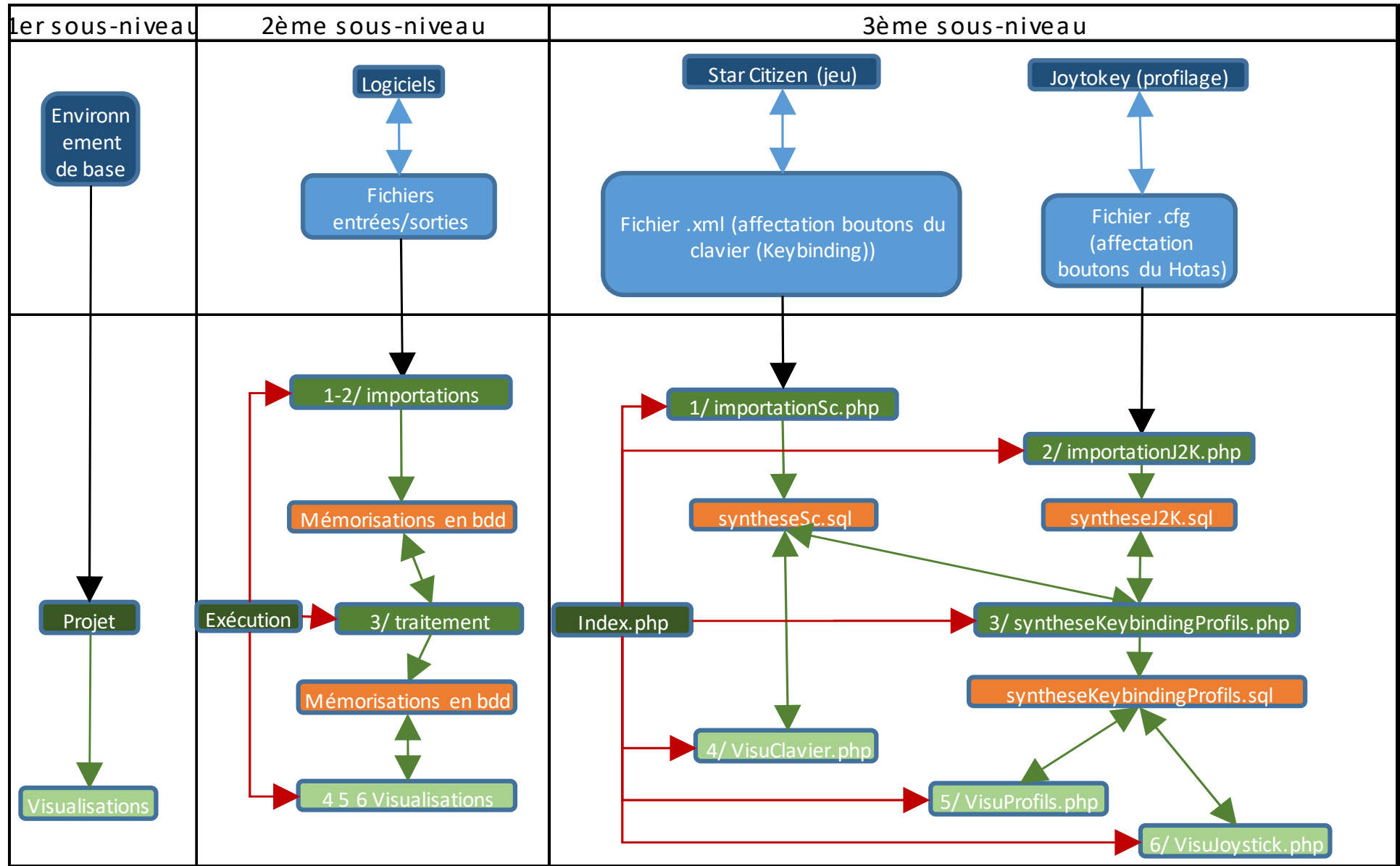
8/ VisuClavier.php

9/ VisuProfils.php

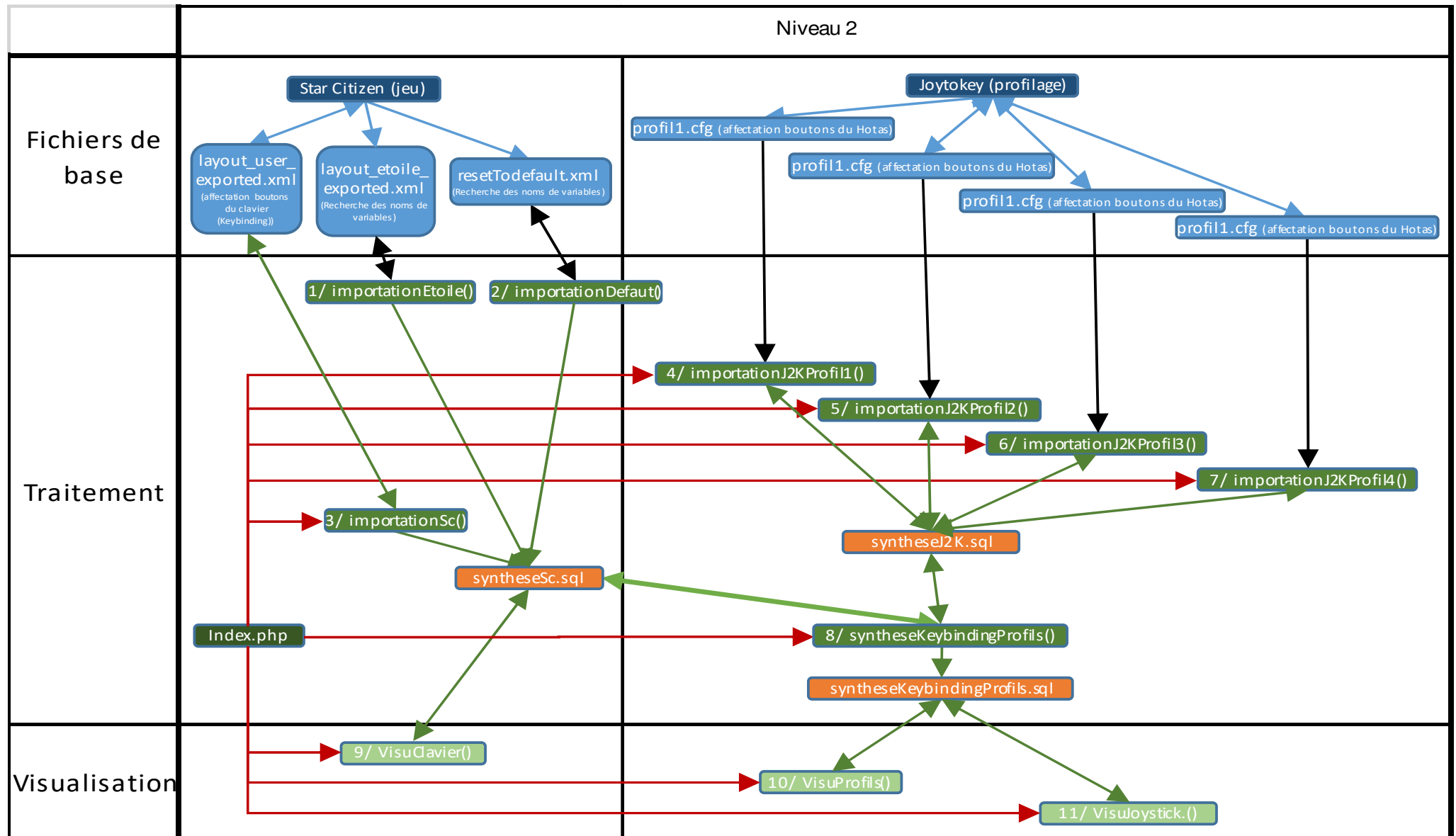
10/ VisuJoystick.php

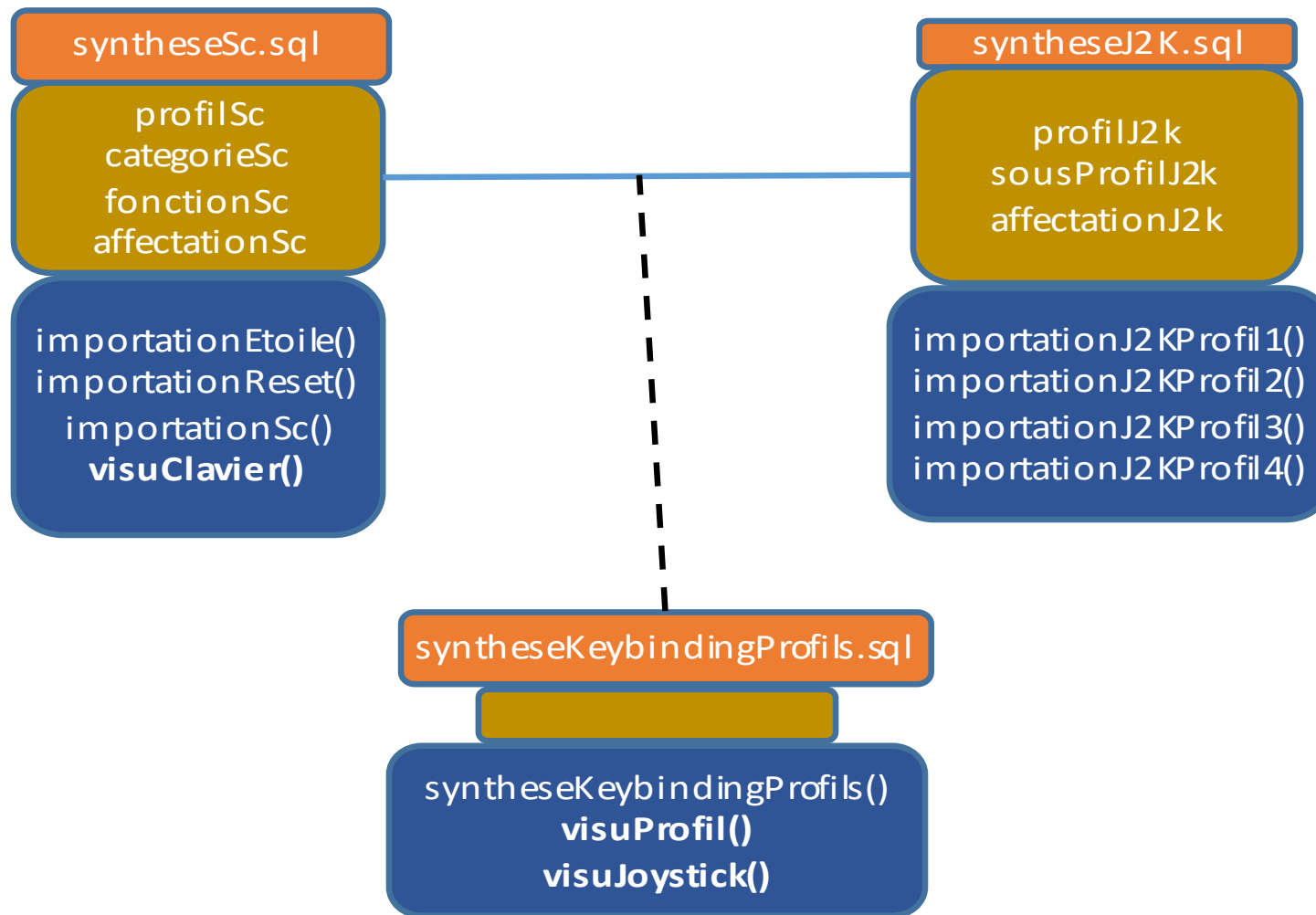
4 Organigrammes

4.1 Organigramme niveau 1



4.2 Organigramme Niveau 2: Adaptabilité au développement du jeu





LA table `syntheseKeybindingProfils` doit pouvoir concaténer la fonction et l'affectation du jeu, avec le profil, le sous profil et l'affectation faite au joystick (JoyToKey).

