

Livrable 3.3.6

Version	1.0
Date	01 septembre 2016
Auteur(s)	R. ALEXIA (ATEME)
N° du Lot	-



Le lecteur media accessible à tous

Livrable 3.3.6 : Outil de conversion de contenus vers le format MPEG-DASH

Titre du projet	media4Dplayer
Abréviation	M4DP
Désignation	media4Dplayer, le lecteur media accessible à tous.
Durée du projet	De Janvier 2015 à Juin 2016 – 18 mois
Coordinateur projet	France Télévisions
Partenaires projet	FRANCE TELEVISIONS (FTV) LE GROUPE LA POSTE (LP) DOTSCREEN (DTS) ATEME (ATM) INSTITUT MINES TELECOM (TSP) LABORATOIRE CHART / LUTIN-USERLAB (UP8) PLAINE COMMUNE (PC)
Prestataires	Multimédia France Production (MFP) Holken Consultants & Partners (HC)
Organisme labellisateur	CAP DIGITAL
Financeurs	La Région Ile-de-France La BPIfrance
Titre de subvention	Fonds Unique Interministériel – FUI18

Le projet media4Dplayer, lecteur media accessible à tous.

Outil de conversion de contenus vers le format MPEG-DASH

Date de soumission : 01/09/2016

Version : 1.0

Objectif(s) du livrable

Code source et documentation des outils de conversion de contenus PLUZZ vers le format MPEG-DASH.

Historique	Date	Modification(s)
V 1.0	01/09/2016	création

Le projet media4Dplayer

Media4Dplayer est un projet collaboratif labellisé par le pôle de compétitivité Cap Digital et subventionné au titre du Fonds Unique Interministériel (FUI) par la région Île de France et BPIFrance. Ce projet de recherche et de développement s'inscrit dans la stratégie de Cap Digital, autour des thématiques d'accessibilité des contenus, de développement numérique et de Silver économie.

Durée de projet 18 mois : Janvier 2015 – Juin 2016

Avertissement

Les informations contenues dans ce document peuvent être sujet à modification sans préavis. Société ou noms de produits mentionnés dans ce document peuvent être des marques ou des marques déposées de leurs sociétés respectives.

Tous les droits sont réservés

Le document est la propriété des membres du consortium media4Dplayer. Aucune copie ou distribution, sous quelque forme ou par tout moyen, n'est autorisée sans l'accord écrit et préalable du (des) propriétaire(s) des droits.

Ce document ne reflète que le point de vue de ses auteurs. Le consortium media4Dplayer et les financeurs ne peuvent être tenus responsables de l'usage qui pourrait être fait des informations contenues dans ce document.

©2016 media4Dplayer

Table Des Matières

1. CONTEXTE DU PROJET	5
2. INTRODUCTION	5
3. OBJECTIF	6
4. OUTILS UTILISES	6
5. PREPARATION DES CONTENUS	6
5.1. RE-PACKAGING DE L'AUDIO	6
5.2. CONVERSION DES FICHIERS TTML:	6
6. PROCEDURE DE CONVERSION D'UN CONTENU	7
7. DEVELOPPEMENT REALISE:	7
7.1. SCRIPT DE GENERATION DE FICHIERS DE CONFIGURATION.....	7
7.1.1. DONNEES EN ENTREE	7
7.1.2. CONFIGURATION.....	8
7.1.3. DONNEES EN SORTIE.....	8
7.1.4. USAGE.....	8
7.2. SCRIPT DE CONVERSION DE FICHIERS MP4 EN CONTENUS DASH EN UTILISANT LES OUTILS GPAC	8
7.2.1. DONNEES EN ENTREE	8
7.2.2. CONFIGURATION.....	8
7.2.3. DONNEES EN SORTIE.....	9
7.2.4. USAGE.....	9
8. EXEMPLE D'UTILISATION DE L'OUTIL	10
8.1. ELEMENTS EN ENTREE	10
8.2. UTILISATION DU SCRIPT GEN-CONF.SH	10
8.2.1. DONNEES EN ENTREE	10
8.2.2. CONFIGURATION.....	10
8.2.3. USAGE.....	10
8.2.1. DONNEES EN SORTIE.....	11
8.1. UTILISATION DU SCRIPT MP42DASH	12
8.1.1. DONNEES EN ENTREE	12
8.1.2. UTILISATION	12
8.1.1. DONNEES EN SORTIE.....	12
9. CONCLUSION	13

1. Contexte du projet

L'objectif du sous-programme 3.3 est la conception et le développement de mécanismes de synchronisation multiflux provenant de serveurs différents. Pour ce faire, plusieurs approches ont été considérées, allant du marquage des flux élémentaires à la définition de mécanismes dédiés aux diverses couches systèmes et transports considérées. L'état d'avancement des technologies Web nous a fait considérer, dès le départ, des développements visant des rendus multi-plateformes (HTML5, IOS, Android).

L'objectif final est d'établir une solution relativement unifiée à même de répondre aux diverses contraintes des architectures des plateformes considérées. Toutefois, les problématiques de synchronisation live (consultation de serveur d'horloge,...) et offline (dérives d'horloges intrinsèques,...) impliqueront naturellement la mise en place de solutions différentes en amont de l'horodatage considéré.

Ces mécanismes ont tout d'abord été mis en œuvre en considérant la technologie de compression H.264/MPEG-4 AVC, cette dernière étant la plus largement déployée dans les écosystèmes de visualisation de contenus sur second écran. Dans un second temps, les travaux de recherche ont été étendus à la nouvelle technologie de compression HEVC (*High Efficiency Video Coding*), capable de performances débit-distorsion deux fois supérieures à H.264/MPEG-4 AVC. Une telle technologie, poussée par le monde de l'OTT (Over The Top), nous permet d'accroître de manière significative l'efficacité d'encodage, et donc d'étendre l'accessibilité aux services.

2. Introduction

Sur la base d'un service principal présenté sur un écran principal connecté (ex : TV via une Set Top Box), de nombreux flux enrichis peuvent être proposés sur un écran secondaire (ex : terminal portable additionnel). Nous pouvons citer par exemple :

- Sondage ou questions à choix multiples relatifs au contenu présenté sur la TV.
- Piste audio alternative écoutée à l'aide d'un casque connecté directement sur le terminal portable en même temps que le flux vidéo principal sur la TV. Il peut s'agir d'informations supplémentaires, de la traduction dans une autre langue, d'un flux d'audio description.
- Visualisation du même contenu que celui présenté sur la TV, mais dans une vue différente.
- Présentation d'informations supplémentaires concernant le contenu principal : informations sur les acteurs au moment où ils apparaissent dans le film, publicité et lien vers un site de vente en ligne d'un produit visible à l'écran.

3. Objectif

Les contenus du serveur **francetvPLUZZ** seront mis à disposition via le player Media4D. Cependant, ces contenus sont disponibles sous forme de quatre fichiers MP4, contenant chacun une piste audio et une piste vidéo, et d'un fichier optionnel de sous-titres au format TTML. Afin de les rendre accessibles par le player, il est nécessaire de les convertir au format MPEG-DASH. Ce format a, en effet, été choisi lors de la phase de spécifications du player.

Les quatre pistes audio sont identiques alors que les pistes vidéo représentent des qualités d'encodage (résolution et débit) différentes.

Aucune étape supplémentaire de transcodage n'est nécessaire. En effet, lors de l'encodage du flux vidéo en H.264, un GOP de taille fixe a été paramétré, permettant ainsi de segmenter les fichiers vidéos obtenus pour la distribution en HTTP Adaptive Streaming. Une unique opération de packaging permettra de produire des contenus au format MPEG-DASH. Cette tâche, décomposée en plusieurs étapes devait pouvoir être automatisée, c'est dans ce but que l'outil spécifié dans ce document a été développé.

4. Outils utilisés

La manipulation des fichiers MP4 est réalisée avec l'outil MP4Box appartenant à la suite logicielle GPAC.

Les scripts d'automatisation ont été développés en shell BASH.

La validation des contenus au format MPEG-DASH est réalisée avec le player javascript Dash.js et avec le player de GPAC MP4Client.

5. Préparation des contenus

La conversion des fichiers MP4 comporte plusieurs étapes spécifiques liées aux spécificités des fichiers audio et de sous-titres.

5.1. Re-packaging de l'audio

L'utilisation directe de l'audio contenu dans les fichiers MP4 produit des contenus qui sont lus sans problème par MP4Client. Cependant, ce mode de packaging au format MP4 n'est pas supporté par Dash.js. La solution identifiée consiste à extraire l'ES audio du fichier MP4 pour le packager dans un nouveau fichier MP4. Ces fichiers, convertis par la suite au format MPEG-DASH, sont supportés par Dash.js et MP4Client.

5.2. Conversion des fichiers TTML:

Les fichiers initiaux de sous-titres TTML utilisent des timestamps au format `hh:mm:ss:frame_num`. Ce format n'étant pas supporté par Dash.js, il est donc nécessaire de les convertir au format `hh:mm:ss.ms`.

6. Procédure de conversion d'un contenu

Cinq étapes distinctes sont nécessaires à la conversion d'un contenu PLUZZ au format MPEG-DASH exploitable par le player dash.js :

- 1 - Génération d'un fichier de configuration contenant :
 - le nom du contenu, aussi utilisé comme répertoire de destination,
 - le profil MPEG-DASH live ou file,
 - la liste des fichiers du contenu, ainsi que le type de média qu'ils contiennent ;
- 2 - Re-packaging de l'audio (MP4Box) ;
- 3 - Conversion des sous-titres TTML ;
- 4 - Import du fichier de sous-titres TTML au format XML dans un fichier MP4, (MP4Box) ;
- 5 - Génération des fichiers MP4 au format MPEG-DASH et du manifest associé.

7. Développement réalisé:

Afin de faciliter l'automatisation de la conversion des contenus, deux scripts ont été développés:

- `gen-conf.sh` : permet la génération automatique des fichiers de configuration de tous les contenus présents dans un répertoire ;
- `mp42dash.sh` : permet la préparation et la conversion d'un contenu PLUZZ en utilisant les informations contenues dans un fichier de configuration généré par le script `gen-conf.sh`.

7.1. Script de génération de fichiers de configuration

Nom	gen-conf.sh
Version	0.3
Description	Script de génération de fichiers de configuration utilisés en entrée du script de conversion de fichiers MP4 en contenus MPEG-DASH
Dépendances	/

7.1.1. Données en entrée

Le script prend entrée un répertoire contenant x contenus PLUZZ à traiter.

Les contenus sont présents dans une arborescence à 2 niveaux « contenu/version » chaque répertoire contient 4 fichiers .mp4 (*-*-standard1.mp4, *-*-standard4.mp4, *-*-standard3.mp4, *-*-standard4.mp4) contenant une piste vidéo (trackID 3) et au moins une piste audio (trackID 4) et optionnellement un fichier de sous-titres (*.ttml).

7.1.2. Configuration

Deux variables sont à configurer avant l'utilisation du script :

- répertoire racine des contenus à analyser : `root_src_dir="../../PLUZZ"`;
- répertoire de destination des fichiers de configuration : `config_dst_dir="dashify-config"`.

7.1.3. Données en sortie

Pour chaque contenu, un fichier de configuration est généré dans le répertoire courant et est nommé d'après le chemin du contenu `dashify_-_contenu_-_version.txt`.

7.1.4. Usage

- 1 - Configurer le chemin vers les répertoires des contenus à analyser et le répertoire de destination des fichiers de configuration ;
- 2 - Exécuter le script `gen-conf.sh` sans arguments.

7.2. Script de conversion de fichiers MP4 en contenus DASH en utilisant les outils GPAC

Nom	<code>mp42dash.sh</code>
Version	0.3
Description	Script de conversion de fichiers MP4 en contenus MPEG-DASH.
Dépendances	MP4Box de GPAC

7.2.1. Données en entrée

Le script prend en entrée :

- un fichier de configuration généré par le script `gen-conf.sh` ;
- le répertoire de contenu correspondant.

7.2.2. Configuration

Trois variables sont à configurer avant l'utilisation du script, les deux premières sont liées à l'environnement d'exécution et la troisième est une option de la packétisation MPEG-DASH :

- répertoire des bibliothèques GPAC : `lib_dir="/opt/gpac/lib"`;
- chemin vers le binaire MP4Box : `bin="/opt/gpac/bin/MP4Box"`;
- durée de segment MPEG-DASH en millisecondes : `segment_duration=2000`.

7.2.3. Données en sortie

Le contenu MPEG-DASH est généré dans le répertoire destination en fonction du nom indiqué dans le fichier de configuration

7.2.4. Usage

Exécuter le script `mp42dash.sh` avec les arguments :

- `-c <file>`: fichier de configuration à utiliser ;
- `-d <dir>`: répertoire de destination racine ;
- `-r` : désactive le re-packaging de la track audio ;
- `-t` : désactive la conversion des timestamps TTML.

8. Exemple d'utilisation de l'outil

Voici un exemple complet de l'utilisation des deux scripts afin de préparer un contenu PLUZZ en vue de sa mise à disposition au format MPEG-DASH.

8.1. Éléments en entrée

Les fichiers MP4 audio, vidéo et sous-titres non segmentés et encodés avec des profils d'encodage adaptés à la distribution en HAS.

8.2. Utilisation du script gen-conf.sh

8.2.1. Données en entrée

Le contenu du répertoire de média /video/136520832/56df564315b46-INFRAROUGE :

```
136520832-56df564315b46-1457485204.ttml.xml
136520832-56df564315b46-1457485204.ttml.xml.mp4
136520832-56df564315b46-qaq.mp4
136520832-56df564315b46-qad.mp4
136520832-56df564315b46-standard1.mp4
136520832-56df564315b46-standard2.mp4
136520832-56df564315b46-standard3.mp4
136520832-56df564315b46-standard4.mp4
136520832-56df564315b46-standard5.mp4
```

8.2.2. Configuration

```
root_src_dir="/video"
config_dst_dir="dashify-config"
```

8.2.3. Usage

```
> gen-conf.sh
```

8.2.1. Données en sortie

Le fichier de configuration correspondant `./dashify-config/dashify_-_infrarouge.txt` :

```
# nom du répertoire destination
infrarouge

# profil MPEG-DASH live ou file
live

# fichier vidéo
/video/136520832-56df564315b46-INFRAROUGE/136520832-56df564315b46-
standard5.mp4#video

# fichier vidéo
/video/136520832-56df564315b46-INFRAROUGE/136520832-56df564315b46-
standard4.mp4#video

# fichier vidéo
/video/136520832-56df564315b46-INFRAROUGE/136520832-56df564315b46-
standard3.mp4#video

# fichier vidéo
/video/136520832-56df564315b46-INFRAROUGE/136520832-56df564315b46-
standard2.mp4#video

# fichier audio et vidéo, vidéo sur la track 3
/video/136520832-56df564315b46-INFRAROUGE/136520832-56df564315b46-
standard1.mp4#trackID=3#video

# fichier audio et vidéo, audio sur la track 4
/video/136520832-56df564315b46-INFRAROUGE/136520832-56df564315b46-
standard1.mp4#trackID=4#audio

# fichier audio
/video/136520832-56df564315b46-INFRAROUGE/136520832-56df564315b46-qa
a.mp4#audio

# fichier audio
/video/136520832-56df564315b46-INFRAROUGE/136520832-56df564315b46-q
ad.mp4#audio

# fichier sous-titres
/video/136520832-56df564315b46-INFRAROUGE/136520832-56df564315b46-
1457485204.ttml.xml#subtitle
```

8.1. Utilisation du script mp42dash

8.1.1. Données en entrée

- le contenu du répertoire `/video/136520832/56df564315b46-INFRA ROUGE` ;
- le fichier de configuration `./dashify-config/dashify_-_infrarouge.txt` généré à l'étape précédente par le script `gen-conf.sh`.

8.1.2. Utilisation

```
/mp42dash.sh -c ./dashify-config/dashify_-_infrarouge.txt -d /dash
```

8.1.1. Données en sortie

Le répertoire destination nommé `/dash/infrarouge-dash-file` contenant :

- les segments MPEG-DASH des différentes représentations (audio, vidéo et sous-titres) ;
- le manifest (`.mpd`) associé.

9. Conclusion

L'outil présenté dans ce document permet l'automatisation de la préparation des contenus PLUZZ afin de les rendre compatible avec le player Media4D qui supporte principalement le format MPEG-DASH. Les développements ont tout d'abord été validés lors des tests unitaires sur une plateforme de locale, puis dans des conditions « réelles » sur une plateforme France TV.