

## Livrable 3.3.3

Version	1.0
Date	15 mai 2016
Auteur(s)	R. ALEXIA (ATEME)
N° du Lot	-



Le lecteur media accessible à tous

## Livrable 3.3.3 : Encodage et horodatage multi-plateformes « offline »

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

## Le projet media4Dplayer, lecteur media accessible à tous.

### *Encodage et horodatage multi-plateformes « offline »*

Date de soumission : 15/05/2016

Version : 1.0

#### Objectif(s) du livrable

Description de l'encodage et horodatage visant un rendu multi-plateformes sur second écran, intégrant les problématiques de synchronisation de multi-flux offline

Historique	Date	Modification(s)
V 1.0	15/05/2016	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 BPI France. 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.....	6
2.	INTRODUCTION .....	6
3.	SYNCHRONISATION MULTI-FLUX INTER-ECRANS.....	7
3.1.	MULTIPLE FLUX DASH .....	8
3.2.	MULTIPLE ECRANS DE LECTURE.....	11
4.	CONCLUSION .....	12
5.	REFERENCES .....	13

## FIGURES

Figure 1 :	Architecture réseau du service de streaming vidéo OTT .....	8
------------	---	---

## TABLEAUX

Tableau 1 : Contenus mis à disposition du projet collaboratif.....	10
--	----

## 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.

Tous ces cas d'usage nécessitent une synchronisation des contenus présentés sur les deux écrans. Les exigences en termes de synchronisation varient selon les types de contenus. La valeur de 40ms est généralement utilisée dans le cas d'un flux vidéo et d'un flux audio (synchronisation labiale) ou de deux flux vidéo (synchronisation à l'image).

L'analyse des différents cas d'usage adressés dans le cadre du projet Media4DPlayer (Cf. [2]) nous a permis de mettre en évidence les modes de synchronisation à considérer.

La première phase des travaux réalisés sur ce projet s'est attachée à gérer les problématiques liées à la synchronisation des flux enrichis au sein d'un seul et même écran (Cf. [3]). En effet, ce premier cas d'usage a permis de répondre aux besoins de synchronisation des différentes instances de players au cours de la lecture, tout en s'affranchissant des contraintes inter-écrans et transmission Live dans un premier temps.

Nous avons pu mettre en œuvre ces travaux à travers une chaîne d'encodage type File sur des flux au format MPEG-DASH, et valider ainsi l'architecture de notre solution technologique.

Dans la phase suivante du projet media4Dplayer, nous nous sommes intéressés aux cas d'usage orientés second écran, ou « companion screen », qui mettent en avant des exigences de synchronisation inter-terminaux.

### 3. Synchronisation multi-flux inter-écrans

La synchronisation de flux multiples provenant de différents serveurs, ainsi que le couplage de la diffusion d'un programme sur un écran dit "principal" avec un second écran (Smartphone, tablette, notebook...) sont les défis qu'il nous faut relever.

Le premier nécessite un horodatage précis des différents flux lors de leur fabrication, d'une façon efficace et compatible avec les cibles considérées, pour ensuite les resynchroniser côté client. Ce point technique est adressé à travers le support du protocole MPEG-DASH.

Le second relève d'un tout autre défi, puisqu'il n'existe aucun lien entre l'écran principal et le second. Ce point est à l'étude, et devrait être satisfait par la mise en œuvre de la technique d'empreinte numérique *fingerprinting*.

### 3.1. Multiple flux DASH

Dans le cadre de notre projet media4Dplayer, les différents médias (vidéo, audio, ...) relatifs à un même contenu sont disponibles dans des flux MPEG DASH distincts (*manifest*), et stockés sur un serveur HTTP.

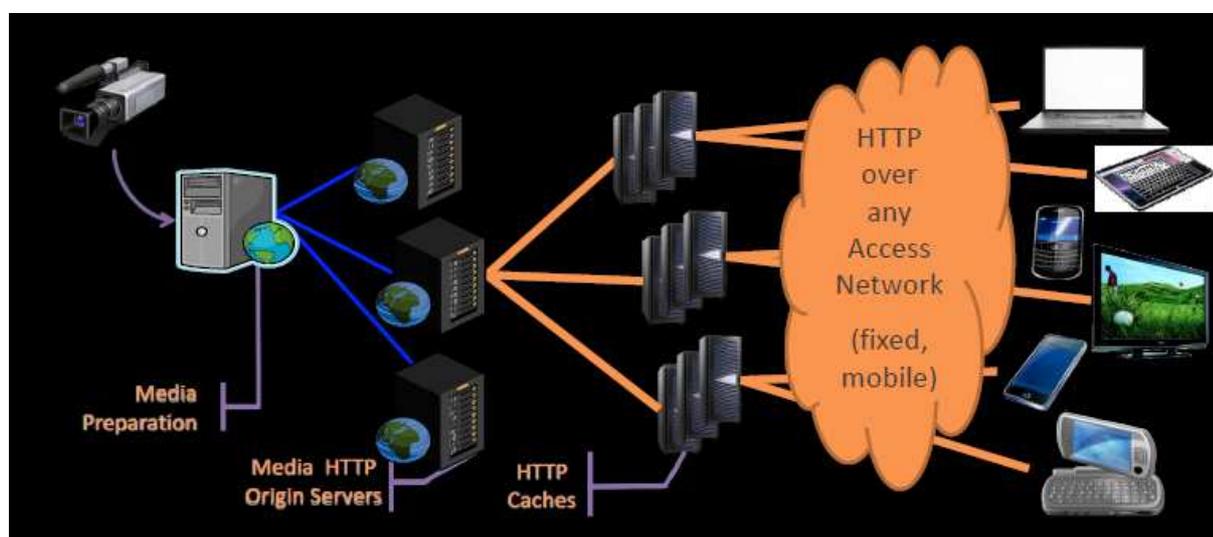


Figure 1 : Architecture réseau du service de streaming vidéo OTT

Afin de recomposer le contenu au niveau du player, l'agrégation et la présentation de ces flux nécessitent de synchroniser le téléchargement (mise en buffer) et la lecture de flux provenant de sources différentes.

Conservant comme nécessité d'être ouvert au plus grand nombre de plateformes, la contrainte de disposer d'un lecteur compatible HTML5 nous a mené vers une première solution : l'utilisation du module « *MediaController* » permettant de synchroniser des flux MPEG-DASH. D'autres propositions sont également à l'étude (framework *Popcorn.js*) afin de viser une meilleure interopérabilité.

L'architecture cible retenue est donc un player *DASH.js* avec une instanciation de la fonction *MediaController* disponible avec le navigateur Chrome sur PC (écran principal) ou tablette (*deuxième écran*) sous OS Windows.

Ainsi, nous mettons à disposition sur un serveur de *francetélévisions* des encodages de programmes enrichis, décrits par les caractéristiques suivantes :

- *Nom-du-programme.mxf* :
  - Vidéo principale : H.264, Full HD 3.6Mbps / HD 2.4Mbps / SD 1.6Mbps
  - Audio principale, HE AAC v1, 64kbps, stéréo
- *Nom-du-programme\_LSF.mov* :
  - Vidéo LSF,
- *Nom-du-programme\_AD.wav* :
  - Audio description, HE AAC V1, 64kbps, mono
- *Nom-du-programme\_EA1.aif* :
  - Audio étendue (multicanal français) HE AAC V1 , 192kbps, 5.1
- *Nom-du-programme\_EA2.aif* :
  - Audio étendue (version multilingue), stéréo
- *Nom-du-programme\_EA3.aif* :
  - Audio étendue (version internationale) HE AAC V1, 192kbps, 5.1 ou stéréo
- *Nom-du-programme\_DI.aif* :
  - Dialogues français, HE AAC V1, 64Kbps, mono
- *Nom-du-programme\_ST1.stl* :
  - Sous-titres français à convertir au format TTML,
- *Nom-du-programme\_ST2.stl* :
  - Sous-titres anglais à convertir au format TTML,
- *Nom-du-programme\_ST3.stl* :
  - Sous-titres (autres...) à convertir au format TTML,

Ces informations sont lues sur demande par les players à travers les *manifests* MPEG-DASH suivants :

- manifest.mpd :
  - Vidéo principale
  - Audio principal
  - Sous-titres 1, 2, 3
- manifest-lsf.mpd :
  - Vidéo LSF
- manifest-ad.mpd :
  - Audio description
- manifest-ea1.mpd :
  - Audio étendu 1
- manifest-ea2.mpd :
  - Audio étendu 2
- manifest-ea3.mpd :
  - Audio étendu 3
- manifest-di.mpd :
  - Dialogues séparés
- manifest-audio.mpd :
  - Audio description

- Audio étendu 1, 2, 3
- Dialogues séparés

L'objectif de cette répartition des contenus est de valider la bonne répartition des différents flux dans les *manifests* vis-à-vis des players DASH.

Ci-dessous la liste des contenus mis à disposition pour les essais collaboratifs (sous <http://videos-pmd.francetv.fr/innovation/media4D/>) :

PROGRAMME	ÉPISODE	TYPE	FORMAT (SD/HD/UHD)
LE MONDE DE JAMY [01]	Des volcans et des hommes	Documentaire	HD
ALEX HUGO	Comme un oiseau sans ailes	Série	HD
RUGBY [01]	France / Angleterre	Sport	HD / AVC
CE SOIR OU JAMAIS		Plateau	HD
DES CHIFFRES ET DES LETTRES		Jeux TV	HD
MÉTÉO		Météo	HD
JOURNAL DE 20h00		News	HD
TCHOUPI	Un gentil ballon [S01.E06]	Dessin Animé	HD
RUGBY [02]	France / Angleterre	Sport	HD / HEVC

Tableau 1 : Contenus mis à disposition du projet collaboratif

Dans le cadre du « *companion screen* », nous étudions en parallèle les plateformes Android et MacOS qui demeurent cependant plus complexes, de par leur évolutivité performante mais incessante pour la première, ou leur compatibilité technologique très restreinte pour la seconde.

## 3.2. Multiple écrans de lecture

Le *fingerprinting* est une technique qui consiste à fabriquer une empreinte numérique (une "image", la plus légère possible) du contenu multimédia sur lequel le second écran pourra se synchroniser.

Cette étape intervient typiquement avant ou en parallèle de la diffusion d'un programme. L'application cliente sur second écran se met ensuite à l'écoute du programme diffusé et exploite un algorithme de comparaison d'empreintes, la première étant mise à disposition sur le web, la seconde fabriquée en temps réel pendant l'enregistrement du signal audio par le microphone intégré du smartphone, de la tablette, de l'ordinateur portable...

Le mécanisme de calcul en déduit une information de décalage temporel, puis prend en compte l'instant de l'enregistrement et le temps écoulé pour :

- piloter le lecteur multimédia du second écran ;
- se positionner temporellement dans le flux ou le fichier ;
- jouer les contenus additionnels en synchronisation avec le contenu diffusé sur le premier écran.

L'IRCAM, sous-traitant du partenaire francetélévisions, étudie la faisabilité d'une synchronisation second écran multi-plateformes, avec les standards ouverts que sont HTML5, l'API Web Audio, le protocole MPEG-DASH, et souhaite étendre l'expérience audio enrichie avec les interfaces existantes.

Ces travaux sont repris dans le cadre du lot 3.4, charge pour la chaîne d'encodage du lot 3.3 de ne pas altérer ces empreintes numériques afin qu'elles soient facilement identifiables du côté du lecteur.

## 4. Conclusion

Les expérimentations réalisées dans le cadre des problématiques de synchronisation ont révélé des verrous technologiques fortement liés à la maturité des environnements utilisés.

En effet, nos travaux sur les différentes plateformes n'ont pas pu progresser de manière homogène. Les évolutions des navigateurs Web, des *operating system*, ou des players DASH nous ont contraints à remettre en cause nos analyses, et parfois à limiter temporairement notre champ d'action.

Cependant, nos études n'ont jamais été mises en défauts, et nous avons pu éprouver des solutions *offline* (type vidéo à la demande, ou télévision de rattrapage) permettant de consolider notre objectif initial.

Pour la dernière phase de ce projet, nous devons nous aborder les cas d'usage orientés « live », et ses exigences spécifiques de synchronisation inter-terminaux.

## 5. Références

- [1] *Proposed Exploration of “Uniform Signaling for Timeline Alignment”*. Thomas, Emmanuel, Le Feuvre, Jean et Singer, David. 2014. ISO/IEC JTC1/SC29/WG11 N14644.
- [2] *M4DP\_D3.3.1\_Etat\_de\_l\_art\_synchronisation\_fine\_v1.0*. ATEME. Mars 2015, media4Dplayer
- [3] *M4DP\_D3.3.2\_Spécifications\_des\_mécanismes\_de\_synchronisation\_v1.0*. ATEME. Septembre 2015, media4Dplayer