



Université
de Toulouse

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut National Polytechnique de Toulouse (INP Toulouse)

Discipline ou spécialité :

Image, Information et Hypermédia

Présentée et soutenue par :

M. NICOLAS BERTRAND

le jeudi 3 décembre 2015

Titre :

TRANSITION VERS LE CINEMA NUMERIQUE : CONTRIBUTIONS
POUR LE CINEMA D'ART ET ESSAI

Ecole doctorale :

Mathématiques, Informatique, Télécommunications de Toulouse (MITT)

Unité de recherche :

Institut de Recherche en Informatique de Toulouse (I.R.I.T.)

Directeur(s) de Thèse :

M. JEAN DENIS DUROU

Rapporteurs :

Mme LUCE MORIN, INSA DE RENNES

M. OGE MARQUES, FLORIDA ATLANTIC UNIVERSITY

Membre(s) du jury :

M. VINCENT CHARVILLAT, INP TOULOUSE, Président

M. JEAN DENIS DUROU, UNIVERSITE TOULOUSE 3, Membre

M. JEROME BRODIER, GNCR, Membre

M. JOSE MOURE, UNIVERSITE PARIS 1, Membre

M. RODOLPHE VILLAGE, CINEMA UTOPIA, Membre

M. SAMUEL THIBAUT, UNIVERSITE BORDEAUX 1, Membre

Remerciements

Je tiens tout d'abord à remercier Rodolphe Village, qui est à l'origine de ce projet et m'a accompagné tout le long de cette thèse. Merci à Anne-Marie Faucon, Michel Malacarnet ainsi que toutes les équipes des cinémas Utopia pour m'avoir permis de travailler dans leur superbes cinémas et pour m'avoir guidé dans l'univers de l'exploitation cinématographique.

Je remercie la Mairie de Tournefeuille et le Syndicat des Salles de Cinéma des Pyrénées pour leurs contributions financières à la réalisation de cette thèse.

Mes remerciements vont aussi à Jean-Denis Durou, directeur de thèse, qui m'a accompagné et conseillé durant cette thèse, et fut, en particulier, d'une vigilance orthographique à toute épreuve. Je remercie Vincent Charvillat, pour m'avoir accueilli dans son équipe et pour m'avoir fourni des idées éclairantes.

Je remercie les associations FFDN et Tetaneutral.net, notamment Laurent Guerby et Julien Rabier, pour la mise à disposition des infrastructures, et surtout, un grand merci pour les pistes techniques données.

Je remercie l'association VideoLAN, notamment Jean-Baptiste Kempf, pour m'avoir suivi dans le dédale de l'application VLC et des codecs libres.

J'exprime tous mes remerciements à Luce Morin et Oge Marques pour avoir bien voulu être rapporteurs de ce travail de thèse. Je remercie Samuel Thibault, qui m'a fourni de précieuses remarques techniques, ainsi que José Moure pour avoir apporté sa vision artistique du cinéma à cette soutenance. Je remercie Jérôme Brodier d'avoir participé à la soutenance, et avec lequel le travail de cette thèse va se poursuivre au sein de l'association Cinémascop. Je tiens aussi à remercier tout le public de la soutenance qui a bien voulu participer, en direct, à l'évaluation des projections.

Enfin, un grand merci à Marjorie, Romane, Gaëtan, mes parents et aux copains qui m'ont supporté et motivé tout le long de ce travail.

Résumé

Les salles de cinéma sont passées à l'ère numérique. L'objectif de cette thèse est d'étudier le déploiement du cinéma numérique pour les salles d'art et essai et de proposer des solutions techniques ouvertes, basées sur des logiciels libres et gratuits. Nous présentons les impacts du passage au numérique pour les distributeurs et les exploitants indépendants. Nous détaillons le fonctionnement du cinéma numérique : de la création de contenu au format DCP, au système de projection en salles. Dans cette thèse, nous proposons deux solutions : une solution de projection basée sur le lecteur multimédia VLC, et un système de transmission dématérialisée de DCP des distributeurs vers les exploitants.

Abstract

Movie theaters have switched to the digital era. The objective of this thesis is to study the deployment of digital cinema for arthouse cinemas and to propose open technical solutions based on free and open source software. We present the impacts of the digital switch for independent distributors and exhibitors. We detail the workflow of the digital cinema specification: from creation of DCP content to the projection system in theaters. This thesis offers two solutions: a projection system based on the VLC media player, and a dematerialized transmission system of DCP from distributors to exhibitors.

Table des matières

1	Introduction	13
2	L'exploitation cinématographique à l'ère numérique	17
2.1	Introduction	17
2.2	La chaîne industrielle du cinéma	17
2.3	Transition vers le numérique	19
2.3.1	Financement par VPF	21
2.3.2	Aide à la numérisation des salles par le CNC	23
2.3.3	Prochain renouvellement du matériel de projection	23
2.3.4	Projection en relief	24
2.4	Spécification du DCI pour le cinéma numérique	24
2.4.1	<i>Master</i> numérique (DCDM)	25
2.4.2	Compression du <i>master</i> numérique	26
2.4.3	Copie numérique (DCP)	27
2.4.4	Transport des DCP	31
2.4.5	Équipement des salles de cinéma	31
2.4.6	Projection	33
2.4.7	Sécurité	34
2.5	Réponses proposées	34
2.5.1	Projection dédiée	35
2.5.2	Distribution de contenu dédiée	39
2.6	Bilan	43
3	Contributions au lecteur multimédia VLC	45
3.1	Introduction	45
3.2	Choix d'un lecteur multimédia	45
3.3	Présentation de VLC	46
3.3.1	Architecture de VLC	46
3.3.2	Écriture d'un module dans VLC	48
3.3.3	Organisation du code dans VLC	49
3.4	Module DCP de VLC	50
3.4.1	Structure d'un DCP	51
3.4.2	Choix du type de module	51

3.4.3	Bibliothèque ASDCPlib	52
3.4.4	Implémentation du module DCP	52
3.4.5	Décodage des contenus	54
3.4.6	Rendu des sous-titres	54
3.4.7	Changement d'espace colorimétrique	54
3.5	Évaluation	55
3.5.1	Validation du module DCP dans VLC	55
3.5.2	Évaluation de la lecture d'un DCP	58
3.6	Bilan	60
4	Un décodeur JPEG2000 dédié au cinéma numérique	61
4.1	Introduction	61
4.2	Le format JPEG2000	62
4.2.1	Découpage en tuiles et changement d'espace colorimétrique	62
4.2.2	Transformation en ondelettes discrète	62
4.2.3	Quantification des coefficients d'ondelettes	65
4.2.4	Codage entropique des <i>code blocks</i> (<i>Tier-1</i>)	65
4.2.5	Création du fichier binaire (<i>Tier-2</i>)	65
4.2.6	Décompression du format JPEG2000	66
4.3	Contraintes du cinéma numérique sur le JPEG2000	66
4.3.1	Contraintes sur la transformation en ondelettes	67
4.3.2	Contraintes sur la compression	68
4.3.3	Contraintes sur la formation du train de bits	68
4.4	Solutions de décodage du JPEG2000	68
4.4.1	Décodage matériel du JPEG2000	69
4.4.2	Décodage logiciel du JPEG2000	69
4.4.3	Décodage avec pertes	71
4.5	Description du décodeur proposé	72
4.6	Évaluation	73
4.6.1	Évaluation du décodeur	73
4.6.2	Évaluation du décodage avec pertes	76
4.7	Bilan	78
5	Contributions à la sécurisation des contenus	79
5.1	Introduction	79
5.2	La sécurité dans le cinéma numérique	79
5.2.1	Chiffrement AES	81
5.2.2	Chiffrement RSA	81
5.2.3	Fichier KDM	82
5.3	Chiffrement des DCP	83
5.3.1	Génération de KDM	84
5.3.2	Déchiffrement d'un DCP	86
5.4	Évaluation	87
5.4.1	Évaluation du chiffrement de DCP	87

5.4.2	Évaluation de la lecture de DCP chiffré avec VLC	88
5.5	Bilan	89
6	Un réseau opérationnel de distribution de DCP	91
6.1	Introduction	91
6.2	Deux notions clés pour la distribution de DCP	92
6.2.1	Fonctionnement d'un CDN	92
6.2.2	Protocole BitTorrent	93
6.3	Présentation du CDN ED-CINE	94
6.3.1	Procédure de routage des requêtes	95
6.3.2	Modèle mathématique du CDN ED-CINE	96
6.4	Conception d'un nouveau CDN	98
6.4.1	Description des besoins	98
6.4.2	Comparaison avec l'architecture du CDN ED-CINE	99
6.4.3	Fonctionnement du système de distribution	101
6.5	Mise en œuvre du CDN The DCP Bay	103
6.5.1	Interfaces logicielles de supervision	103
6.5.2	Logiciels exploitant l'infrastructure matérielle	104
6.5.3	Infrastructures	105
6.5.4	Équipement chez les exploitants	107
6.6	Évaluation	108
6.6.1	Évaluation du VPN Tinc	109
6.6.2	Évaluation du client BitTorrent	110
6.6.3	Évaluation globale de The DCP Bay	110
6.7	Vers une optimisation du CDN The DCP Bay	111
6.7.1	Analyse de l'existant	111
6.7.2	Contraintes sur l'optimisation de la distribution	113
6.8	Bilan	113
7	Conclusion et perspectives	115
A	Acronymes	119
	Bibliographie	123

Chapitre 1

Introduction

*Tous les discours sur le numérique,
c'est n'importe quoi! Il faudrait tout
reprendre point par point.*

Jean-Luc Godard

Le principe du cinématographe des frères Lumière est resté le même : la fixation d'images sur un film et leur projection sur un écran par un projecteur, et ce à travers l'évolution des technologies, du muet au parlant, du noir et blanc au Technicolor.

Ce système centenaire est aujourd'hui désuet. La projection cinématographique est passée au numérique, tout comme le livre ou l'appareil photo. L'industrie cinématographique dématérialise les bobines. Le tirage de films est remplacé par la copie de bits sur un disque dur ou la transmission de ces mêmes bits par les réseaux. Les projecteurs 35 *mm* prennent place dans les musées, mais il est fort probable que nous parlerons toujours d'aller voir un « film ».

Les premières projections cinématographiques en numérique ont eu lieu au début des années 90 avec le projecteur *Image Light Amplifier* (ILA) de la société Hughes/JVC. La qualité d'image était très éloignée de celle d'un projecteur 35 *mm*. La projection de films numériques en salle avec une qualité satisfaisante a été véritablement lancée par la technologie *Digital Light Processing* (DLP) développée par la société Texas Instrument. Le premier projecteur pour le cinéma utilisant la technologie DLP est apparu en 1999. Cette technologie permet la projection d'une image par des miroirs microscopiques disposés sous forme de matrice dans un semi-conducteur. Depuis les années 2000, une fois la technique de projection numérique mise au point, tous les aspects concernant la diffusion et la projection de films numériques sont étudiés. La *Society of Motion Picture and Television Engineers* (SMPTE) est un organisme international qui développe des standards pour les industries du cinéma et de l'audiovisuel. En 2000, elle crée le comité DC28, afin de mener une étude sur le cinéma numérique. Ce comité est divisé en sept groupes, qui travaillent sur toute la chaîne de distribution du cinéma, de la création du *master* à la projection en salle. Deux ans plus tard, afin d'accélérer l'adoption du cinéma numérique, les sept *majors* de Hollywood créent un groupement d'experts du cinéma

appelé *Digital Cinema Initiatives* (DCI). Cet organisme est chargé de créer un standard pour le cinéma numérique qui, une fois établi, retournera au sein de la SMPTE, puis de le soumettre comme norme internationale *International Organization for Standardization* (ISO). En 2005, le DCI édite la première version de sa spécification : la *Digital Cinema System Specification* (DCSS). Dix ans plus tard, en France, quasiment toutes les salles de cinéma sont équipées de matériel numérique.

La DCSS retranscrit l'architecture système et les exigences désirées par les studios de Hollywood. Ce document, qui couvre un large spectre, s'appuie sur les documents produits par le DC28. Il décrit le format pour les *masters* et les copies numériques, ainsi que le matériel pour la diffusion des films et la projection en salle. La DCSS est pensée pour la diffusion massive de *blockbusters*, un contrôle strict de la distribution et la sécurisation des copies et des salles.

Ce modèle de distribution provoque des inquiétudes et des interrogations pour le cinéma d'Art et Essai. Le modèle proposé ne semble pas s'adapter aux besoins de ce type de cinéma. Le matériel numérique est bien plus coûteux que l'ancien matériel, et les pratiques d'utilisation sont modifiées. Lors de la transition vers le numérique, les petites salles et les salles d'Art et Essai ont cherché des solutions de financement pour pouvoir s'équiper en matériel compatible avec la DCSS, sans se soucier, dans un premier temps, des conséquences de cette mutation technologique. Les contraintes posées par la spécification du DCI ne sont pas seulement économiques. Elles changent également la manière de diffuser et de projeter, en ajoutant des restrictions sur la période de projection d'un film et des contraintes liées aux dispositifs anti-piratage. Elles remettent également en cause le métier de projectionniste, qui est remplacé par des services d'assistance informatique.

Nous avons étudié des solutions adaptées au cinéma d'Art et Essai. La situation financière des distributeurs de films et des exploitants de salles de cinéma est bien plus fragile que celle des grands groupes de l'exploitation cinématographique. Au regard des normes proposées par le DCI, nous nous sommes demandé si les solutions proposées actuellement étaient en adéquation avec les attentes de ce type d'exploitation.

Le principal problème posé aux exploitants par la transition vers le numérique est financier : le matériel numérique est bien plus coûteux que les projecteurs argentiques. Ce constat est d'autant plus marquant pour les salles indépendantes : le nombre de sièges par projecteur étant bien inférieur à celui d'un multiplexe, l'investissement sur un nouveau projecteur peut d'autant plus affecter l'équilibre financier d'un cinéma.

Cette thèse a été réalisée dans le cadre d'un contrat de type Conventions Industrielles de Formation par la REcherche (CIFRE) avec les cinémas Utopia, qui sont des cinémas indépendants d'Art et Essai. Son objectif principal est de proposer des solutions pour l'exploitation cinématographique, dans le domaine du logiciel libre.

Notre première contribution consiste à proposer une solution conçue pour du matériel standard, qui permette la lecture de copies numériques de films en conformité avec les exigences du DCI. Dans ce cadre, nous avons développé un module dans le lecteur mul-

timédia *VideoLAN Client* (VLC), qui permet de projeter des copies numériques. Nous avons également implémenté un décodeur JPEG2000, qui est le format de compression des images choisi par le DCI, dans la bibliothèque multimédia libre libavcodec. Le code développé a été intégré officiellement dans VLC et libavcodec. Le système de projection que nous proposons est donc accessible librement et gratuitement. En raison de l'utilisation de matériel standard, il peut être connecté à tous types de projecteurs, et donc utilisé pour effectuer des projections de copies numériques au format spécifié par le DCI.

La programmation des salles d'Art et Essai est plurielle et variée. En proportion du nombre de sièges, une salle d'Art et Essai projette plus de films qu'un multiplexe. Ces salles ont donc besoin de recevoir de nombreuses copies de films. Depuis le passage au numérique, les films sont expédiés soit par disque dur, soit de manière dématérialisée par les réseaux Internet. Les offres existantes de transport dématérialisé ne sont pas adaptées à la distribution de films d'Art et Essai, car elles sont trop chères pour le distributeur et pour l'exploitant. Les films d'Art et Essai étant distribués en beaucoup moins d'exemplaires que les *blockbusters*, cela limite l'intérêt du transport dématérialisé, dont le but est de réduire les coûts de transfert.

Notre deuxième contribution est un système dématérialisé de distribution de copies numériques réalisé en partenariat avec des fournisseurs d'accès associatifs, qui s'appuie lui aussi sur des technologies en logiciel libre. Nous avons mis en place un *Content Delivery Network* (CDN) utilisant le principe du « pair à pair » (*peer-to-peer*). Ce système est d'ores et déjà opérationnel et en production. Depuis septembre 2014, soit en un peu plus d'un an, nous avons distribué 250 films sur un réseau auquel 70 écrans sont connectés.

Dans ce mémoire, nous présentons tout d'abord, dans le chapitre 2, le contexte de l'exploitation cinématographique, le fonctionnement du cinéma numérique et les solutions apportées aux problèmes de la projection et de la distribution de copies numériques.

La mise en œuvre du système de projection, qui s'appuie sur le lecteur multimédia VLC, est détaillée dans le chapitre 3. Nous avons développé un décodeur JPEG2000 dédié au cinéma numérique, les problèmes posés et les choix pour l'écriture de ce décodeur étant décrits dans le chapitre 4. Dans ce même chapitre, nous évaluons la qualité d'image et les performances du décodeur.

La sécurisation des contenus est un sujet primordial pour le DCI. Dans le chapitre 5, nous démontrons qu'il est possible de mettre en œuvre une solution sûre et libre de chiffrement et de déchiffrement de copies numériques pour le cinéma.

Les copies numériques pour le cinéma sont des conteneurs de grande taille : de 150 Go à 500 Go par copie. La taille élevée de ce format de fichier constitue le principal obstacle au déploiement d'un système de type CDN pour le cinéma numérique. Dans le chapitre 6, nous proposons un système opérationnel de distribution de copies numériques, appelé « The DCP Bay ». Nous y détaillons l'architecture réseau du système, ainsi que les choix de logiciels libres permettant de rendre ce CDN efficace pour la distribution. Nous évaluons les difficultés rencontrées lors de sa mise en œuvre. Enfin, notre système de distribution est comparé à un système de distribution déjà existant.

Chapitre 2

L'exploitation cinématographique à l'ère numérique

2.1 Introduction

Notre étude se situe dans le domaine de l'exploitation cinématographique. Nous ne sommes pas dans le cadre de la réalisation d'œuvres, mais dans leur exploitation une fois qu'elles ont été réalisées. L'exploitation consiste à distribuer (diffuser) un contenu. Il existe plusieurs possibilités pour diffuser un film : sortie en salles, sortie en DVD/Blu-ray, vidéo à la demande, . . .

Nous nous positionnons dans le domaine de la distribution d'œuvres vers les salles. Une œuvre peut être diffusée à l'échelle locale, nationale ou internationale. Chaque pays a ses propres réglementations pour la diffusion de contenus cinématographiques. Nous nous intéressons ici à l'exploitation cinématographique en France. Même si les moyens techniques de distribution et de projection d'un film sont plus ou moins indépendants du pays, les réglementations peuvent différer.

En France, la création et l'exploitation d'œuvres cinématographiques est régie par le Centre National du Cinéma et de l'image animée (CNC). Cet organisme public a été créé après la Seconde Guerre Mondiale. Le mode de financement du CNC est fondé sur un système de répartition qui vise à financer la création de films, mais aussi à aider les salles de cinéma à se développer. Lors de la vente d'un billet de cinéma, une partie du montant est reversée au CNC.

2.2 La chaîne industrielle du cinéma

L'exploitation cinématographique fait intervenir trois types d'acteurs :

Le producteur – Il finance la réalisation du film. Depuis l'avènement du numérique, l'activité du producteur est distincte de celle du post-producteur. Le post-producteur est chargé des activités techniques : effets spéciaux, colorimétrie, réalisation du *master* (édition de la version finale).

Le distributeur – Il est l’intermédiaire entre le producteur et l’exploitant. C’est lui qui négocie le nombre de films distribués par zone géographique (généralement par pays), ainsi que la durée pendant laquelle le film sera projeté dans une salle. Il existe de « grands » distributeurs, comme les *majors* américaines (qui sont souvent à la fois producteurs et distributeurs), mais aussi des distributeurs français comme Gaumont-Pathé. Il existe également des « petits » distributeurs qui ne sortent que quelques films par an, en général spécialisés (films étrangers, de jeunesse, d’animation).

L’exploitant – Son rôle est de montrer le film au public. Là encore, il existe de nombreux types d’exploitation, du multiplexe en périphérie des grandes villes à la salle mono-écran de campagne. Les exploitations peuvent être destinées au grand public, plus spécialisées dans les films dits d’« Art et Essai », ou bien dans des « films de répertoire ». De plus, toutes les salles de cinéma ne programment pas tous les films au moment de leur sortie nationale. Certains exploitants sont spécialisés dans la « continuation », c’est-à-dire la diffusion quelques semaines (généralement cinq ou six) après la sortie nationale. Et bien sûr, certaines salles mélangent tous ces critères (films grand public, d’Art et Essai, de répertoire, sorties nationales, continuation, ...).

Le travail présenté dans ce mémoire a été financé par les cinémas Utopia, au travers de l’association Indépendants, Solidaires et Fédérés (ISF). Les cinémas Utopia forment un réseau de cinq sites en France (Avignon, Bordeaux, Montpellier, Saint-Ouen et Toulouse/Tournefeuille). Ces cinémas sont indépendants, c’est-à-dire qu’ils ne dépendent ni d’institutions publiques, ni des grands groupes de l’exploitation cinématographique. Ils possèdent tous le label Art et Essai.

Qu’est ce que le cinéma d’Art et Essai ? En France, le classement Art et Essai d’une salle est défini ainsi par le CNC : « *Le classement Art et Essai a pour objectif de soutenir les salles de cinéma qui exposent une proportion conséquente de films recommandés Art et Essai et qui soutiennent ces films souvent difficiles par une politique d’animation adaptée* ».

Les films recommandés Art et Essai constituent une liste de films fournie par l’Association Française des Cinémas d’Art et d’Essai (AFCAE), qui possède ses propres critères¹ sur la qualité artistique d’un film. La liste inclut tous les films ayant obtenu le label « Recherche et Découverte » défini par le CNC, mais on trouve dans cette liste des films à plus grande diffusion, comme *Kill Bill* de Quentin Tarantino. Le spectre des films Art et Essai est donc large, tout comme celui des lieux d’exploitation.

Le label Recherche et Découverte marque la visibilité dans la programmation des œuvres singulières et novatrices du paysage cinématographique. Les œuvres possédant ce label sont souvent soutenues par le Groupement National des Cinémas de Recherche (GNCR). C’est avec le GNCR que nous avons mis en œuvre le projet de transfert dématérialisé de films (cf. chapitre 6).

1. Critères de l’AFCAE : <http://www.art-et-essai.org/11/criteres-de-recommandations-des-oeuvres>

Notre travail s'adresse en fait plutôt aux distributeurs et exploitants qui distribuent et projettent des œuvres ayant obtenu le label Recherche et Découverte, et dont l'équilibre financier est souvent fragile. Le cinéma numérique proposé par les *majors* américaines n'a pas été pensé à l'origine pour ce genre d'œuvres.

Les cinémas Utopia, qui exploitent beaucoup de films Recherche et Découverte, sont un cas d'école pour ce type de programmation. On peut considérer le cinéma Utopia de Toulouse/Tournefeuille comme un « grand » cinéma indépendant : avec sept salles, il compte en moyenne 450 000 entrées par an depuis dix ans.

Utopia s'est équipé en matériel numérique en 2012. Parallèlement à cette transition, le projet de recherche de notre étude a été lancé. La projection en 35 *mm* était une chose connue et maîtrisée techniquement. En revanche, la projection numérique implique de nouvelles technologies électroniques et informatiques. L'équipement actuel est une boîte noire sur laquelle le projectionniste ne peut plus intervenir (il doit faire appel à une assistance téléphonique pour chaque problème rencontré). Utopia s'est donc inquiété, car il courait le risque d'une perte d'indépendance, en devenant étroitement lié à l'installateur du nouveau matériel. C'est pour cette raison que Utopia a souhaité, parallèlement au changement de matériel, initier une recherche sur des solutions libres pour l'exploitation cinématographique.

2.3 Transition vers le numérique

« *Par ailleurs, le cinéma est une industrie* », c'est ainsi qu'André Malraux conclut son essai *Esquisse d'une psychologie du cinéma* consacré à l'esthétique du cinéma. « *Contrairement à la plupart des autres arts, le cinéma puise son essence même dans le lien inextricable entre la création et l'argent* » ([19], page 7).

La principale raison de la transition de l'industrie du cinéma vers le numérique est économique ([30], page 21). Le coût du tirage d'une copie sur bobine 35 *mm* est élevé, et il n'est pas rare qu'un film soit distribué à raison de 4000 copies à sa sortie. Une copie numérique (voir table 2.1) est bien moins coûteuse. En revanche, le coût des équipements numériques est bien plus élevé que celui du 35 *mm*. Le distributeur fait donc des économies, au détriment de l'exploitant. De ce constat a découlé la mise en place de propositions de financement des équipements de salles, à travers les économies réalisées par les distributeurs (cf. paragraphe 2.3.1).

	35 <i>mm</i>	Numérique
Coût d'une copie	600 € à 2 000 €	150 € à 300 €
Coût de l'équipement d'une salle	13 000 € à 25 000 €	75 000 € à 80 000 €

TABLE 2.1 – Comparaison des coûts entre équipement 35 *mm* et équipement numérique. Les variations de coût d'une copie 35 *mm* dépendent du nombre de tirages. Cela est également vrai, dans une moindre mesure, pour une copie numérique [29].

Les principales étapes qui vont de la création du *master* à la projection restent pratiquement inchangées pour le cinéma numérique (voir figure 2.1), certaines pouvant

disparaître en raison de la dématérialisation des copies numériques. Actuellement, la majorité des copies numériques sont livrées par le distributeur aux exploitants sous la forme de disques durs, mais la transmission de copies par le réseau devrait remplacer les échanges matériels.

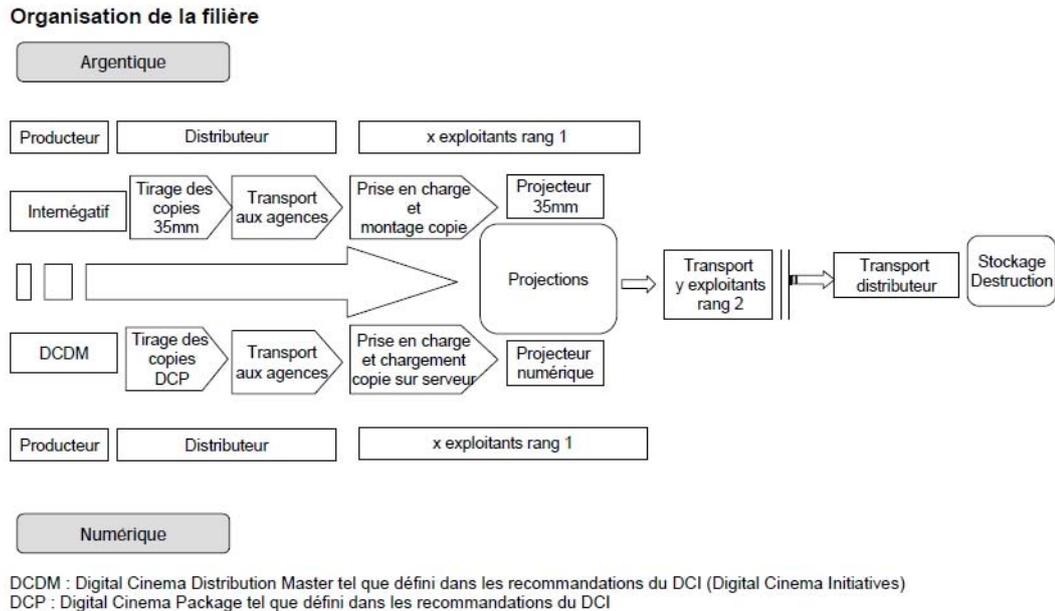


FIGURE 2.1 – Les acteurs de la filière cinématographique (source [29]).

La projection numérique permet aussi la diffusion en direct d'événements sportifs ou culturels dans les salles de cinéma. Un enjeu important soulevé par les nouvelles technologies est celui de la liberté de programmation des salles et de la maîtrise des plans de sortie des distributeurs ([19], page 110). L'auteur souligne que « *les exploitants perçoivent le numérique comme un moyen accru de centralisation dans la commercialisation des films associée [...] à l'élargissement des projections à d'autres spectacles* ». Les salles de cinéma pourraient ainsi se transformer en « espaces de projection ». Les distributeurs craignent aussi la déprogrammation de leurs films au profit d'autres spectacles en direct, durant les moments de grande fréquentation comme le samedi soir.

Selon le CNC, à la fin 2014, 100 % des écrans français étaient numériques. La table 2.2 montre l'évolution de la numérisation en France. La numérisation a été lancée en 2008 et 2009 par le groupe CGR, suivi des autres grands groupes comme UGC ou Gaumont. Les grands exploitants peuvent plus facilement signer des accords avec des distributeurs ou des tiers investisseurs, grâce au mécanisme des VPF (cf. paragraphe 2.3.1), pour financer l'équipement des salles, car ils disposent de beaucoup plus d'écrans. Pour les salles indépendantes ou mono-écran, cela s'est fait un peu plus tard (à partir de 2011-2012) grâce au plan d'aide du CNC.

	Salles	%	Établissements	%
Juin 2007	57	1 %	38	1,8%
Juin 2008	442	8 %	136	6,6%
Juin 2009	930	16,9 %	298	14,5%
Juin 2010	1176	21,4 %	365	17,7 %
Juin 2011	2686	48,9 %	688	33,5 %
Mars 2012	4105	73,7 %	1142	55,2 %
Mars 2013	5177	93,6 %	1751	80,9 %
Mars 2014	5482	97,7 %	1927	94,7 %
Décembre 2014	5653	100 %	2019	100 %

TABLE 2.2 – Évolution de l'équipement des salles en matériel numérique en France. Source CNC (<http://www.cnc.fr/web/fr/1174>).

Au travers de ce plan d'aide, le nombre d'établissements ayant fermé en raison de la transition vers le numérique a été maîtrisé. Le nombre de fermetures est difficile à estimer : en 2012, il y avait 2035 établissements, contre 2019 en 2015, mais certains établissements ont ouvert durant cette période.

Au niveau européen, le pourcentage de salles équipées est globalement très inférieur à celui de la France. Sur la figure 2.2, le pourcentage de salles numériques est représenté par pays. Cette figure est issue du rapport 2013 de Media Salles [2]. Trois pays sont équipés à 100 % (Luxembourg, Norvège et Pays-Bas), la France et la Grande-Bretagne étant équipées à plus de 90 %. Suivent l'Allemagne (68 %) et la Russie (67 %). Enfin, les pays d'Europe de l'Est (hors Russie) sont les moins bien équipés. Dans l'ensemble, les pays qui ont bénéficié de plans d'aide ont pu passer au numérique. Comme cela s'est également produit en France, le système des VPF n'a pas suffi à assurer la transition.

2.3.1 Financement par VPF

Le coût du passage au numérique pour les exploitants de salles a amené la filière cinématographique à imaginer des solutions de financement. La solution la plus répandue est celle des *Virtual Print Fee* (VPF) : « *Les distributeurs qui placent une copie numérique chez un exploitant, versent à celui-ci ou à un intermédiaire, un tiers investisseur, un VPF qui correspond à une partie du gain réalisé par le distributeur sur le coût de la copie [...]* Les tiers investisseurs ont donc la mission de transférer partiellement aux exploitants les économies obtenues par les distributeurs afin de les aider à financer les investissements de la projection numérique » ([19], page 108).

En pratique, les cinémas optant pour les financements via les VPF passent par les tiers investisseurs, car ils ne peuvent pas réaliser les investissements en matériel numérique sur leurs fonds propres. Or, les tiers investisseurs sont soit des intégrateurs système (proposant le déploiement de matériel numérique), soit directement des distributeurs ([30], page 209). Il est donc possible que les films pour lesquels un accord VPF

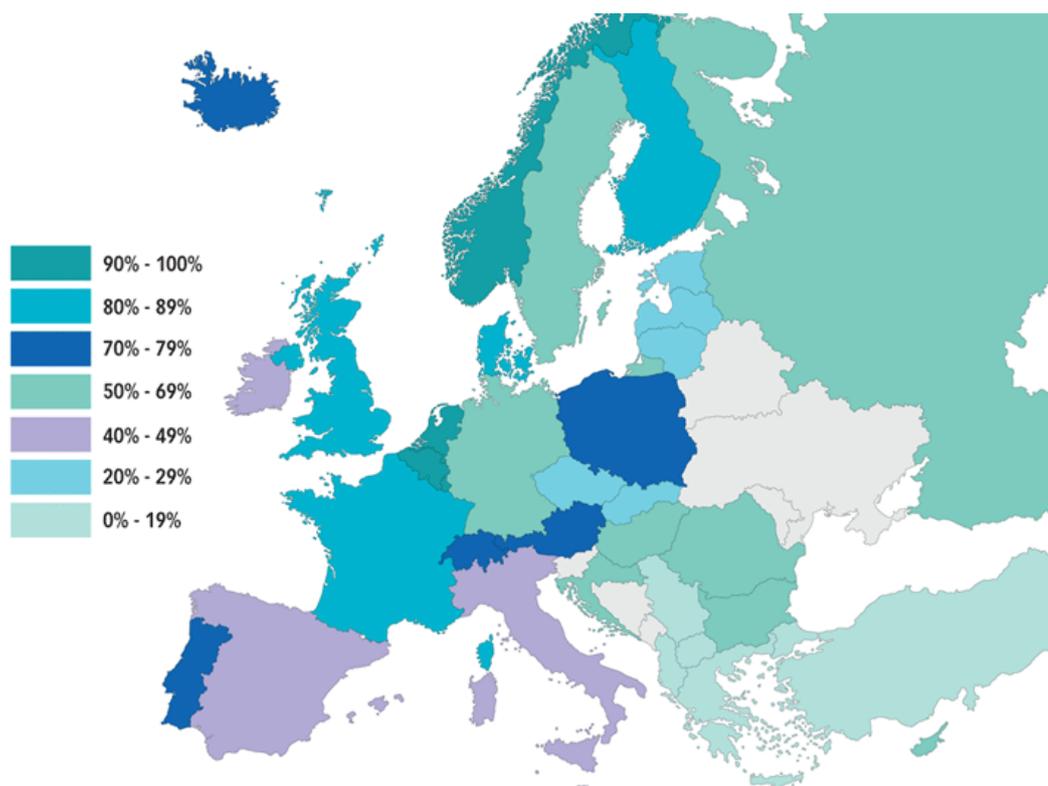


FIGURE 2.2 – Pourcentage de salles numériques en Europe en 2013. Source Media Salles (<http://www.mediasalles.it/ybk2013/index.html>).

a été signé soient, pour des raisons financières, les plus diffusés. Seuls les grands distributeurs ont les moyens de financer les VPF. Il existe donc un risque que les petits distributeurs aient des difficultés à diffuser leurs films, même s'ils sont numériques.

Le rapport [4] de la Commission Européenne relève aussi le risque de « *créer un marché de l'exploitation/de la distribution cinématographique à deux vitesses, où seuls les multiplexes et les films commerciaux à forte valeur marchande bénéficieraient de la révolution numérique du cinéma. Or, la diversité culturelle et le renouvellement des talents dépendent du maintien du réseau cinématographique exceptionnel dont dispose l'Europe* ». Cette hypothèse est levée, en raison des importants investissements dans l'équipement numérique à la charge des exploitants, mais aussi en raison des VPF : « *Il se peut que le modèle VPF constitue une source de revenus insuffisante dans le cas de petits cinémas et de salles d'art et d'essai proposant des films européens. Cette situation peut avoir des répercussions sur la programmation* ». Le cinéma européen est plus morcelé que le cinéma américain : 10 % des cinémas européens sont des multiplexes, contre 35 % aux États-Unis.

Le rapport note que 80 % des écrans européens ont une largeur inférieure à dix mètres et qu'actuellement, il existe des projecteurs dont les caractéristiques sont similaires à celles des projecteurs au format 2K (1920 × 1080, au lieu de 2048 × 1080 pour le 2K). Ces projecteurs dits « e-cinema » sont nettement moins chers que ceux qui sont conformes à la norme DCI. Sur les petits écrans, « *la qualité de projection offerte par une résolution 1920 × 1080 est plus que satisfaisante, et la différence avec le format 2K est totalement acceptable pour le public et les exploitants de salles. Cette solution serait clairement une option viable pour la grande majorité des écrans européens. Elle permettrait en outre de réaliser des économies importantes sur le plan des investissements privés et publics nécessaires pour le passage à la projection numérique* » [4].

Depuis septembre 2010, le CNC propose des solutions de financement complémentaire aux VPF, tout comme l'Europe à travers le programme MEDIA ².

2.3.2 Aide à la numérisation des salles par le CNC

Le plan de numérisation des salles lancé par le CNC est complémentaire du financement par VPF, toutes les salles françaises ne pouvant pas financer le matériel uniquement par VPF. Le plan a aussi pour but d'aider les distributeurs, en numérisant rapidement les salles afin d'éviter les doubles tirages (35 mm et numérique).

« *Le plan de numérisation des salles de cinéma engagé en 2010, afin d'aider les petites salles indépendantes à s'équiper, s'est achevé en 2013. Le comité d'experts de l'aide à la numérisation s'est réuni à cinq reprises, cette dernière année, et a soutenu 372 établissements (538 écrans) pour une aide totale de 23,3 M€. Au total, le CNC a attribué 73,7 M€ pour la numérisation de 1 188 établissements ou circuits itinérants représentant 1 617 écrans ou projecteurs. Le montant moyen de l'aide par établissement s'est élevé à 62 000 €. Les aides attribuées sont composées à 39,5 % (29 M€) de subventions et à 60,5 % (44,7 M€) d'avances remboursables* » (source : rapport d'activité 2013 du CNC, <http://www.cnc.fr/web/fr/publications/-/ressources/5559280>).

2.3.3 Prochain renouvellement du matériel de projection

De nombreuses salles ont commencé à être numérisées en 2010. On estime à dix ans la durée de vie du matériel numérique. Les accords VPF ont une durée de vie de huit ou dix ans. Que se passera-t-il lors de l'obsolescence du matériel en cours d'utilisation ? Il est très probable que les distributeurs ne voudront pas continuer à financer du matériel pour les exploitants. Ce modèle n'est donc pas pérenne.

De plus, le CNC participera-t-il au prochain renouvellement ? Certainement pas avec un plan d'aide exceptionnel, mais de manière plus classique avec le fonds de soutien.

On peut imaginer une baisse de coût du matériel. En 2015, on ne trouve toujours pas d'équipement de salle à moins de 50 000 €. De plus, les fabricants de projecteurs se tournent maintenant vers la projection par laser ³, qui constitue une nouvelle inconnue

2. <http://www.mediafrance.eu/>

3. <http://www.manice.org/projection-laser/principes-de-base-de-la-projection-laser.html>

pour les salles. On peut à la fois craindre un équipement plus coûteux, et espérer faire des économies, car il n’y aura plus de lampe de projecteur à changer.

C’est une des raisons pour lesquelles nous menons cette étude : anticiper le renouvellement des projecteurs numériques et proposer une solution de projection de remplacement, adaptée aux moyens financiers des petites salles.

2.3.4 Projection en relief

La technologie numérique offre la possibilité de projeter en 3D, comme le film *Avatar* de James Cameron. « *Les exploitants sont d’autant moins incités à investir que les gains de qualité de projection numérique par rapport à une copie 35 mm sont loin d’être établis pour le spectateur [...] Pour vaincre ces réticences, les majors américaines ont axé le “marketing” de la projection numérique autour de la 3D* » ([19], page 109). Une copie numérique reste de qualité constante, quel que soit le nombre de projections. Un spectateur regardant un film sur une copie 35 mm ayant déjà servi plusieurs fois verra la différence de qualité.

En revanche, la projection en relief n’est pas encore normalisée. On trouve deux procédés : à lunettes actives ou à lunettes passives. Pour le spectateur, la projection en 3D utilisée comporte encore des défauts : images plus sombres (à cause des filtres polariseurs), migraines à la fin des séances de films d’action⁴, ... De plus, on assiste à une baisse de fréquentation des films 3D. Pour un même film sortant en 2D et en 3D, la proportion de films vus en 3D est en constante régression depuis la sortie d’*Avatar*.

Des réalisateurs comme James Cameron demandent une évolution des projecteurs numériques pour améliorer la 3D, car cette technologie ne leur semble pas encore suffisamment mature.

La projection de films en 3D représente, en 2014, 41 films sur 663 nouveaux films, soit 6 %, ce qui montre que la 3D au cinéma est réservée aux grandes sorties de *blockbusters*. Nous n’avons donc pas pris en compte la projection de films en 3D dans notre travail.

2.4 Spécification du DCI pour le cinéma numérique

Le DCI a été créé par les *majors* de Hollywood : MGM⁵, Disney, Fox, Paramount, Sony Pictures Entertainment, Universal et Warner Bros. Le but principal du DCI est : « *establish and document voluntary specifications for an open architecture for digital cinema that ensures a uniform and high level of technical performance, reliability and quality control*⁶ ».

Le document DCSS [16] spécifie des exigences non précisées par la norme ISO. Cette spécification est souvent appelée « norme DCI ». La dénomination de norme est abusive, car elle ne fait partie d’aucun organisme de normalisation comme la SMPTE ou l’ISO.

4. <http://www.slate.fr/story/3499/1a-3d-fait-mal>

5. La MGM a quitté le DCI en 2005, avant la publication de la première version de la spécification DCSS.

6. <http://www.dcmovies.com/>

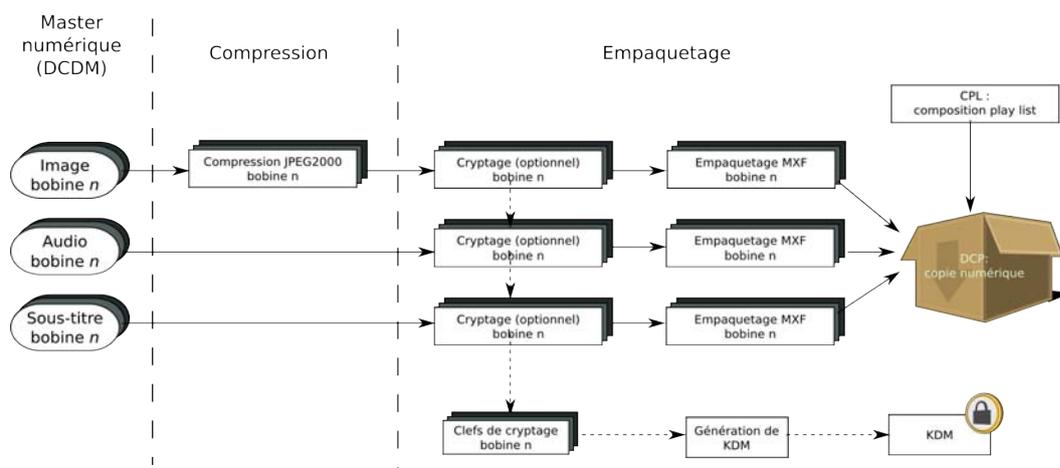


FIGURE 2.3 – Du *master* numérique à la copie numérique.

Il est indiqué, à la page 2 de [15], que ce document « *est conçu uniquement comme un guide pour les entreprises intéressées par le développement de produits, qui peut être compatible avec d'autres produits* ». Le groupe de travail DC28 du SMPTE a standardisé son travail en tant que norme ISO.

La DCSS fait de nombreuses références à ces standards. Le document comporte sept parties : *master* numérique, compression, empaquetage, transport, équipement des salles, projection et sécurité. Ces différents points sont détaillés dans les sous-chapitres ci-après.

La DCSS introduit la notion de « *composition* ». Elle représente toutes les données nécessaires à la réalisation d'un long métrage, d'une publicité ou d'une bande-annonce. Les données de ces différents supports sont similaires (images, son, sous-titres). Nous utiliserons cette notion par la suite pour décrire tous types d'œuvres.

Le flux de données permettant de passer du *master* numérique, appelé *Digital Cinema Distribution Master* (DCDM), à la copie numérique, appelée *Digital Cinema Package* (DCP), est représenté sur la figure 2.3. Les étapes de ce flux sont décrites ci-dessous.

2.4.1 *Master* numérique (DCDM)

Le DCDM est un ensemble de fichiers représentant une *composition*. Cet ensemble comprend des fichiers d'images, de son et de sous-titres. Le *master* est généré une fois que la phase de post-production est terminée. C'est le point d'entrée du flux de données du cinéma numérique.

Une *composition* pour le cinéma numérique est composée de séquences d'images *Tagged Image File Format* (TIFF) de 16 bits de profondeur, où seuls les 12 bits de poids fort sont utilisés. On encode les couleurs dans l'espace colorimétrique CIE 1931 XYZ, qui se fonde sur un modèle de perception visuelle. La spécification du cinéma numérique est le premier format numérique à directement utiliser cet espace colorimétrique, qui est décrit plus en détail ci-après.

La fréquence des images est variable, de 24 images par seconde, fréquence la plus courante, à 60 images par seconde pour la technologie *High Frame Rate* (HFR), qui est peu utilisée. Deux types de tailles d'images sont spécifiés : le 2K (2048 × 1080 pixels) et le 4K (4096 × 2160 pixels).

Les fichiers de son sont encodés au format *Waveform Audio-file Format* (WAV), à une fréquence de 24 *kHz* ou 48 *kHz*.

Les sous-titres peuvent être codés dans des fichiers au format *Portable Network Graphics* (PNG), ou bien sous forme de texte, au format *eXtensible Markup Language* (XML).

Espaces colorimétriques

Les couleurs des images sont définies dans l'espace colorimétrique CIE XYZ. La SMPTE définit la « fonction de transfert » suivante :

$$\begin{aligned} X' &= \text{INT} \left[4095 \times \left(\frac{X}{52,37} \right)^{1/2,6} \right] \\ Y' &= \text{INT} \left[4095 \times \left(\frac{Y}{52,37} \right)^{1/2,6} \right] \\ Z' &= \text{INT} \left[4095 \times \left(\frac{Z}{52,37} \right)^{1/2,6} \right] \end{aligned} \quad (2.1)$$

Les valeurs « tri-stimulées » XYZ sont divisées par la constante 52,37, qui permet de normaliser la luminance Y entre 0 et 1. Les formules (2.1) fournissent donc, avant conversion en entiers, des valeurs comprises entre 0 et $4095 = 2^{12} - 1$. Par conséquent, après conversion en entiers, les valeurs X'Y'Z' peuvent être codées sur 12 bits.

L'espace colorimétrique XYZ, qui est linéaire par définition, représente des niveaux de luminance qui sont proportionnels à la lumière sur l'écran. L'encodage linéaire de la lumière est pratique pour les images de synthèse, car la physique sous-jacente est linéaire. Cependant, l'encodage linéaire ne correspond pas au système de perception humain qui est approximativement logarithmique [22]. C'est pour cela que l'on retrouve l'exposant 1/2,6 pour l'encodage des valeurs X'Y'Z'. On appelle l'inverse de cet exposant le « gamma ». Le gamma a été fixé à 2,6 pour le cinéma numérique, car cette valeur est utilisée par le projecteur DLP depuis plusieurs années et a également été évaluée par plusieurs coloristes [22].

Le choix de l'encodage sur 12 bits est justifié dans l'annexe E de [43]. Un encodage sur 12 bits possède plus de bits que nécessaire pour éviter toute perte d'information due à la profondeur de bits sur une image ou aux artéfacts d'une image. Ce résultat est obtenu avec la courbe *Human Vision Modulation Threshold* (HVMT), qui est représentée sur la figure 2.4. Cette courbe est issue des travaux de Barten [6].

2.4.2 Compression du *master* numérique

Le DCI a choisi, durant l'été 2004, JPEG2000 comme format de compression des images des copies numériques. En raison de la grande taille des *masters* numériques (de

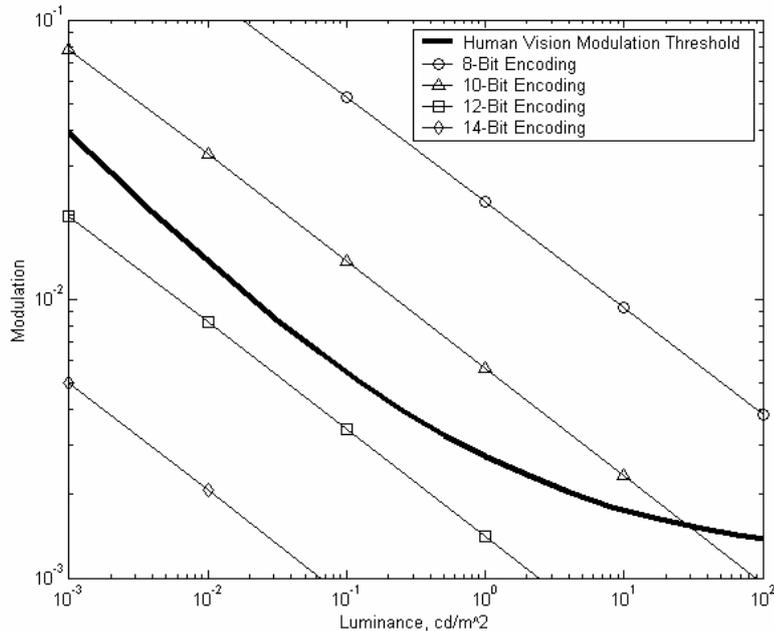


FIGURE 2.4 – Courbe HVMT et visibilité de l’encodage à différentes profondeurs de bits.

2 To à 8 To), la compression des images est indispensable pour permettre la distribution des œuvres. Parmi les données, seules les images sont compressées.

La DCSS spécifie que l’algorithme de compression doit être un standard ouvert, afin que chaque constructeur de matériel puisse construire son système de cinéma numérique. L’algorithme de compression doit supporter une profondeur de bits élevée (12 bits par composante de couleur), et un espace colorimétrique XYZ sans sous-échantillonnage de la chrominance (le même nombre de bits doit être utilisé pour chaque composante, codage 4:4:4). Il doit aussi pouvoir gérer les définitions 2K et 4K à partir du même fichier. Le JPEG2000 satisfait à ces exigences [7].

La publication [37] va plus loin dans les raisons du choix de JPEG2000. L’algorithme de compression doit avoir une licence libre. La compression doit être visuellement sans perte, en conditions normales de projection. Une qualité d’image constante avec un débit variable doit être préférée à un débit constant avec une qualité d’image variable.

Nous reviendrons plus en détail sur la compression JPEG2000 dans le chapitre 4.

2.4.3 Copie numérique (DCP)

Le DCP est le format choisi par le DCI comme conteneur (*container*). Comme dans le format 35 mm, la notion de *bobine* est utilisée. Dans le cas du numérique, cette notion est virtuelle et correspond à une durée arbitraire (entre 20 et 30 minutes).

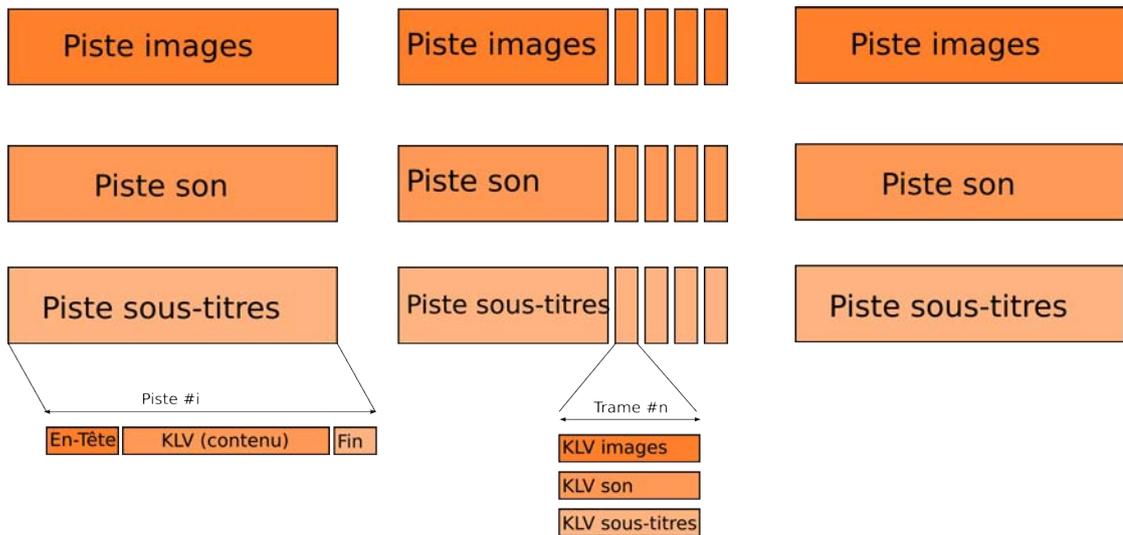


FIGURE 2.5 – Structure d’une copie numérique, de la bobine à la trame. Une copie numérique est composée de plusieurs bobines. Ces dernières contiennent des pistes images, son et éventuellement sous-titres. Chaque piste possède la même structure. Une piste est découpée en trames, qui sont encodées séparément.

Une bobine est composée d’une ou de plusieurs pistes (*track files*). Chaque piste correspond à un type de média : images, son ou sous-titres.

Chaque piste a la structure du conteneur *Material eXchange Format* (MXF) standardisé par la SMPTE. Le MXF est un fichier pouvant regrouper de nombreux contenus comme de la vidéo, du son ou des méta-données. Ce format est fondé sur le codage de données *Key-Length-Value* (KLV), qui lui aussi est standardisé par la SMPTE. Il utilise des triplets composés d’une clé de 16 octets (qui identifie le type de données transportées), de la taille des données et des données [33].

Une piste est découpée en trames (une trame, ou *frame*, correspond à une image) et est codée au format KLV. Les pistes son et sous-titres sont encodées séparément et correspondent aux images (voir figure 2.5).

Les trames codées au format KLV peuvent être chiffrées. Le chiffrement utilise le standard *Advanced Encryption Standard* (AES) et des clés symétriques (clés identiques pour le chiffrement et pour le déchiffrement). La même clé est donc utilisée pour garantir l’authenticité et l’intégrité.

Dans le cas d’une copie chiffrée, l’ensemble des clés utilisées est stocké dans un fichier appelé *Key Delivery Message* (KDM). Ce fichier est lui aussi chiffré avec des algorithmes à clé publique comme *Rivest-Shamir-Adleman* (RSA), c’est-à-dire que ce n’est pas la même clé qui est utilisée pour le chiffrement et pour le déchiffrement.

Dans les salles, chaque équipement de projection possède sa propre clé. Un chiffrement de KDM est réalisé pour chaque équipement, en fonction de sa clé. Cela permet

de s'assurer qu'une copie ne puisse être projetée que dans une salle donnée. De plus, dans un KDM, on trouve une période de validité de la clé symétrique, ce qui permet de contrôler la durée de projection d'un film. Si un exploitant désire projeter le film plus longtemps, il doit demander une nouvelle clé. Il s'agit du moyen mis en place par le DCI pour contrôler les droits d'un film.

Caractéristiques du conteneur DCP

Un conteneur DCP est un répertoire dans lequel on trouve des fichiers XML et des fichiers MXF. La figure 2.6 montre un exemple de structure de DCP. Les fichiers XML suivants contiennent des méta-données sur le DCP :

Volume Index – Contient des informations sur le nombre de volumes sur lequel est stocké le DCP. La taille d'un DCP est généralement élevée, jusqu'à 400 ou 500 Gb. Néanmoins, vu la capacité de stockage des disques durs actuels, un seul volume suffit.

Asset Map – Liste tous les fichiers contenus dans le répertoire du DCP, à part lui-même et le *Volume Index*. L'*Asset Map* fait le lien entre le nom et l'emplacement physique dans le répertoire, et le *Universally Unique Identifier* (UUID)⁷ qui identifie chaque fichier. Dans les autres fichiers, chaque élément (*asset*) est identifié par son UUID.

Packing List – Également appelée *PackIng List* (PKL), elle permet d'identifier tous les éléments contenus dans le DCP, notamment la taille et le code de vérification de chaque fichier (*hash*). Généralement, il n'y a qu'une seule PKL par répertoire de DCP, mais il arrive que des distributeurs ou des laboratoires livrent des DCP avec plusieurs PKL (et donc plusieurs contenus), souvent afin de limiter le nombre d'envois.

Composition List – Également appelée *Composition Play List* (CPL), il peut y en avoir une ou plusieurs par conteneur DCP, par exemple une pour le film en version originale, l'autre pour la version sous-titrée ou la bande son dans une autre langue.

Les fichiers contenant la vidéo et l'audio sont au format MXF. La vidéo est une suite d'images au format JPEG2000. L'audio est codé au format WAV. Les sous-titres sont stockés la plupart du temps au format XML. Ils peuvent aussi être codés sous forme d'images au format PNG. Le format XML est le plus utilisé pour les sous-titres, car il est le plus souple. Pour le format PNG, il faut générer une image pour chaque sous-titre à afficher, ce qui alourdit le conteneur DCP.

Un DCP est souvent découpé en bobines (comme le 35 mm) virtuelles, i.e. en séquences de 20 à 30 minutes. Cela permet d'avoir des fichiers, notamment vidéo, de plus petite taille : les vérifications comme les contrôles d'intégrité sont facilitées, car les fichiers sont plus légers, et les traitements peuvent être faits en parallèle.

7. UUID version 4 (nombres pseudo-aléatoires) : <http://tools.ietf.org/html/rfc4122>.

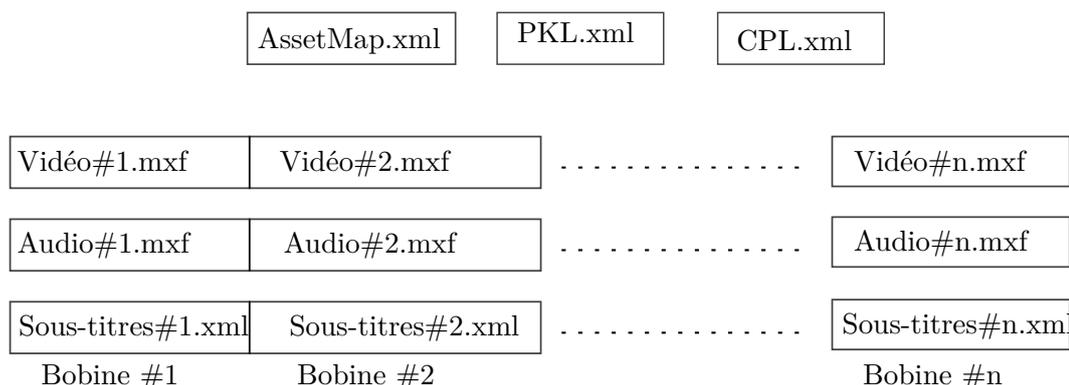


FIGURE 2.6 – Le conteneur DCP est composé de fichiers XML pour les méta-données et de fichiers MXF pour le contenu, sauf pour les sous-titres qui sont au format XML ou PNG, et qui sont stockés dans un sous-répertoire avec la police de caractères associée.

Il peut exister plusieurs DCP pour un même film, par exemple la version originale et une version pour malentendants. Cela pourrait consister en plusieurs CPL, mais ce sont souvent des laboratoires (américains par exemple) qui produisent la version originale, avant qu’un autre DCP soit produit pour les malentendants français (par un laboratoire français, dans ce cas).

Le nommage des DCP respecte une convention définie par l’*Inter-Society Digital Cinema Forum* (ISDCF)⁸. Prenons l’exemple du film *Ombre des Femmes*, qui a donné lieu à trois DCP :

1. `OmbreDesFemmes_FTR-1_S-250_FR-XX_FR_51_2K_SBS_20150427_DGM_IOP_OV`, de taille 95,11 Go.
2. `OmbreDesFemmes_FTR-1_S-250_FR-XX_FR_51-VI_2K_SBS_20150427_DGM_IOP_VF`, de taille 5,06 Go.
3. `OmbreDesFemmes_FTR-1_S-250_FR-FR-OCAP_FR_51_2K_SBS_20150427_DGM_IOP_VF`, de taille 204,7 ko.

Il s’agit des trois DCP pour la première version du long métrage (FTR-1). Le film est au format *scope*, qui définit un rapport d’aspect égal à 2,5 (S-250), information qui permet de choisir l’objectif du projecteur. Le DCP principal est celui qui se termine par OV (*Original Version*), les deux autres étant des compléments de contenu : ils se terminent par VF (*Version File*). Le DCP en OV est en français non sous-titré (FR-XX). Le 51 indique que, pour ce DCP, le son est au format 5.1, le 2K que le film est au format 2K. Dans le deuxième DCP, l’indication 51-VI signifie qu’il s’agit de la version pour déficients visuels (VI pour *Vision Impaired*) : il contient des pistes audio supplémentaires de description. Le dernier DCP possède l’annotation FR-FR-OCAP : il contient des sous-titres supplémentaires pour malentendants, qui seront affichés à l’écran (OCAP pour *Open CAPtion*).

8. Convention de nommage des DCP : <http://digitalcinemanamingconvention.com>

On note dans cet exemple une propriété intéressante du conteneur DCP : seul le fichier original a une grande taille (95,11 Go), car il contient les images, qui ne dépendent pas de la version. On y ajoute soit une piste audio (5,06 Go), soit des sous-titres (204,7 ko). On n'est donc pas obligé de rééditer complètement un DCP pour lui apporter des modifications. Cela évite de manipuler plusieurs fichiers de taille volumineuse.

On retrouve ces fichiers de contenu complémentaire pour ajouter des logos au début du film (par exemple, celui du Festival de Cannes ou celui du distributeur). Pour l'ajout de logos, cette pratique n'est malheureusement pas toujours mise en place par les distributeurs, qui préfèrent générer des DCP « complets », comportant juste l'ajout d'un logo. Cela provoque souvent quelques tensions du côté des exploitants, qui souhaitent recevoir le film à temps pour la première projection. Par exemple, pour le film *Mille et une Nuits*, volume 1, un ajout de logo a été effectué le 17 juin 2015 pour une sortie nationale le 24 juin. La taille du DCP étant de 228 Gb, sa redistribution s'est faite dans la précipitation, alors que le logo ne pesait pas plus de quelques Mo. Envoyer un DCP contenant simplement le logo aurait donc été plus judicieux, mais en revanche, cela n'aurait pas garanti au distributeur que l'exploitant le diffuserait. En effet, s'il s'agit d'un DCP complémentaire, l'exploitant a le choix de l'ajouter ou non à sa liste de lecture.

2.4.4 Transport des DCP

La DCSS ne spécifie pas de stratégie pour la distribution d'une copie du distributeur à l'exploitant : les choix adoptés peuvent être le support physique (disque dur), le satellite ou la transmission filaire à haut débit [32].

D'un point de vue économique, on retrouve ici un phénomène décrit à la page 111 de [19] : l'échec des stratégies de convergence. Les grands distributeurs, comme les *majors* de Hollywood, sont des groupes organisés de manière verticale, de la production à la distribution du film. Ils possèdent également leurs propres réseaux de distribution. Or, ceux qui produisent du contenu souhaitent une large diffusion, pas seulement sur leur réseau. Inversement, les réseaux (au sens des sociétés fournissant des accès aux réseaux) souhaitent diffuser le plus de contenu possible, tout en essayant d'en obtenir l'exclusivité vis-à-vis des réseaux concurrents.

C'est certainement cette raison qui a conduit le DCI à ne pas spécifier le transport de copies numériques. Dans l'introduction de la DCSS, il est écrit que le but du document est de proposer une architecture ouverte pour le cinéma numérique. La non spécification de l'étape de transport va à l'encontre de ce but.

2.4.5 Équipement des salles de cinéma

Un système de projection numérique dans un cinéma comprend tous les équipements nécessaires à la projection dans une ou plusieurs salles de l'établissement :

Serveurs de projection – Les serveurs permettent de stocker, déchiffrer et décoder du contenu (DCP) dans un environnement sécurisé.

Media block – Le *Media Block* (MB) est l'équipement qui déchiffre et décode le contenu. Soit il constitue un équipement séparé, soit il est intégré au projecteur, auquel cas il est appelé *Integrated Media Block* (IMB).

Projecteur – Il permet de projeter l'image décodée sur l'écran de la salle.

Chaîne son – Elle comprend le processeur son, les amplificateurs et les enceintes qui permettent le rendu du son en salle.

Automations – Ce sont les équipements qui permettent de commander l'éclairage, le rideau de scène, ...

Interface utilisateur – Également appelée *Screen Management System* (SMS), elle permet au projectionniste de superviser la projection dans une salle. À travers le SMS, il peut démarrer et stopper des séances, ou préparer des séquences de lecture. Cela ne comprend pas seulement l'ajout de contenu, mais aussi le réglage de l'objectif du projecteur suivant le format du film, les niveaux sonores et les *automations*. Le SMS se situe généralement dans la cabine de projection.

Ces équipements sont dupliqués pour chaque salle, suivant le nombre d'équipements du cinéma.

De plus, il existe des équipements qui centralisent la réception des DCP dans le cinéma. Ce dispositif est nécessaire pour les salles multi-écrans, mais les cinémas mono-écran aussi en sont souvent pourvus. Ces équipements comportent les éléments suivants :

Bibliothèque – C'est un espace de stockage de grande capacité, qui comporte des services actifs pour la réception (*ingest*) de DCP. La copie dans la bibliothèque se fait soit par disque dur externe, soit par des services de transfert de fichiers comme *File Transfer Protocol* (FTP). Depuis la bibliothèque, on peut envoyer des DCP vers les serveurs de projection.

Équipements de réception – Ce sont des boîtiers qui permettent de recevoir des DCP depuis l'extérieur du réseau du cinéma. Il y en a généralement un par transporteur. Une fois le DCP reçu, il peut être envoyé vers la bibliothèque.

Interface utilisateur – Également appelée *Theater Management System* (TMS), elle permet de réceptionner les DCP, de superviser les projections et les équipements. Un TMS comporte toutes les fonctions d'un SMS. Le TMS se situe généralement près de la caisse. Il permet de lancer les projections et de vérifier leur bon déroulement.

Tous les équipements numériques sont reliés à un réseau interne. Ce réseau est divisé en deux sous-réseaux : un pour tout ce qui relève des données de contrôle et de supervision (appelé « réseau de communication »), un autre pour ce qui relève du contenu (appelé « réseau média »). Des routeurs et des *switchs* sont installés pour les interconnexions.

Par exemple, un DCP est reçu sur un équipement de réception, qui le transfère sur la bibliothèque via le réseau média (flèches rouges sur la figure 2.7). Il indique au projectionniste sur le TMS que le DCP est disponible via le réseau de communication. Le DCP est ensuite transféré vers un serveur de projection, via le réseau de communication. En revanche, la clé de déchiffrement du DCP, qui est souvent reçue par courriel, est transmise au projecteur via le réseau de communication (flèches bleues sur la figure 2.7).

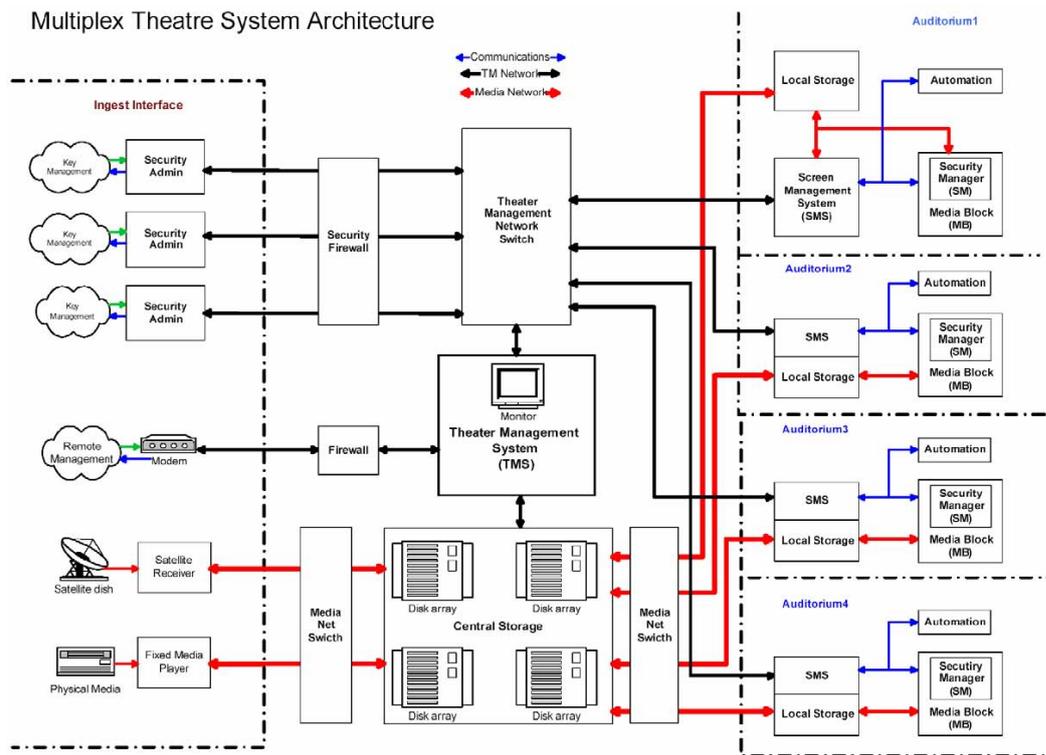


FIGURE 2.7 – Système de projection DCI pour une exploitation multi-écrans (source DCSS [16]).

La figure 2.7 montre l'architecture d'un système de projection pour une exploitation multi-écrans. L'architecture d'une salle mono-écran est équivalente, quoique plus simple : il n'y a généralement pas de TMS, mais on y retrouve souvent une bibliothèque de stockage de DCP séparée du projecteur (appelée *Central Storage* sur la figure 2.7).

2.4.6 Projection

Le projecteur est l'élément le plus important du cinéma numérique. Sa fonction consiste à transformer des images numériques en lumière. La technologie DLP de la société Texas Instrument est la plus utilisée. Sony propose aussi sa propre technologie appelée *Silicon X-tal Reflective Display (SXRD)* et utilisée dans ses projecteurs pour le cinéma numérique. Le projecteur est utilisé au début de la chaîne pour étalonner le *master* numérique et, évidemment, à la fin pour la projection en salle.

La DCSS définit avec précision les caractéristiques techniques d'un projecteur en matière de définition, de contraste, d'uniformité de la couleur, ...

Le MB est l'élément du système chargé du déchiffrement et de la décompression du contenu.

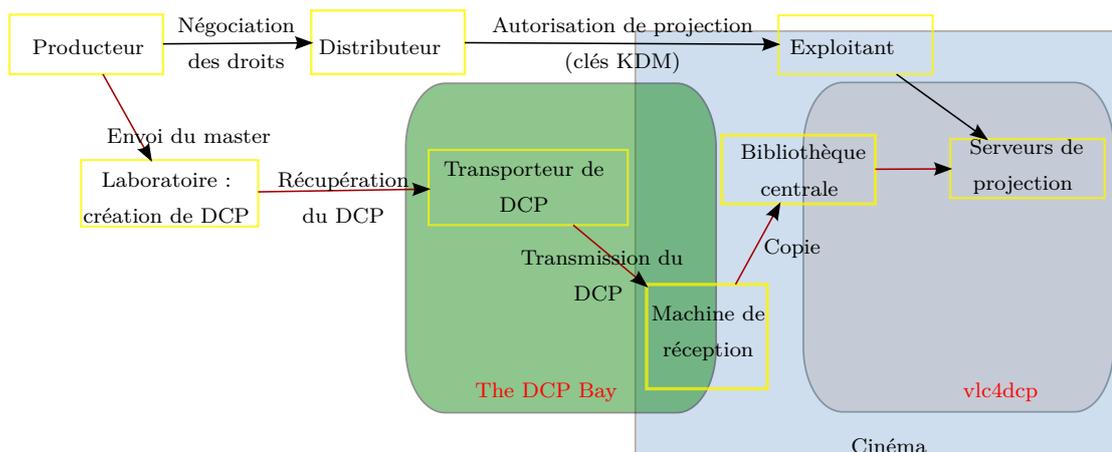


FIGURE 2.8 – Vue globale du cinéma numérique. La figure représente la circulation de DCP des distributeurs vers les exploitants.

Le MB et le projecteur sont connectés par une liaison Dual Link HD-SDI⁹. Le MB est soit intégré physiquement dans le projecteur, solution préférée par le DCI pour des raisons de sécurité du contenu, soit séparé, mais dans ce cas la liaison HD-SDI doit être chiffrée. Le sous-titrage ou les autres affichages graphiques passent par une liaison Ethernet 100Base-T, ainsi que les commandes de contrôle du projecteur (marche/arrêt, choix de l'objectif, ...).

De plus, le film doit être marqué (voir le prochain paragraphe) par le MB, intégré ou non au projecteur, entre le décryptage et l'envoi sur la liaison HD-SDI.

2.4.7 Sécurité

Les méthodes de sécurité mises en place visent à protéger le contenu et à lutter contre le piratage. La transition vers le numérique a amplifié la crainte de vols de films et de leur diffusion illégale (DVD, Internet). Dans la DCSS, de nombreuses mesures de sécurité ont été mises en place. Elles seront présentées dans le chapitre 5.

2.5 Réponses proposées

La figure 2.8 présente le cadre de nos contributions à l'exploitation cinématographique. Nous avons travaillé sur un système de projection appelé « vlc4dcp » et sur un système de transport de copies numériques appelé « The DCP Bay ».

9. Dual Link HD-SDI est une interface numérique, normalisée par la SMPTE (372M) à 1,5 Gb/s, supportant la définition du 2K. Pour le 4K, on ajoute une autre liaison, donc cela devient une liaison Dual-Dual Link HD-SDI.

Notre premier objectif est de proposer une solution alternative de projection dédiée au cinéma d'Art et Essai. Cette solution doit être moins coûteuse que les solutions actuelles de projection conformes à la DCSS. Notre deuxième objectif est de proposer un système de distribution de DCP des distributeurs vers les exploitants, qui soit adapté à des œuvres plutôt confidentielles. Dans les deux cas, nous proposons des solutions basées sur les principes du logiciel libre.

2.5.1 Projection dédiée

Notre but n'est pas de proposer un nouveau système de projection, mais plutôt de fournir un système simplifié qui puisse s'adapter à la projection dans des petites salles (moins de 50 sièges). Cela permet l'ajout d'une petite salle dans un cinéma existant, ou bien l'émergence de petits cinémas de quartier (comme des ciné-clubs) ou de campagne, pour lesquels l'achat de matériel conforme au DCI est prohibitif, mais qui souhaitent tout de même projeter des films au format DCP.

Prenons l'exemple du cinéma Utopia de Montpellier, qui est actuellement équipé de deux salles. Les exploitants souhaiteraient construire une petite salle supplémentaire de 30 places. Cette salle servirait à projeter des films en « continuation », c'est-à-dire six semaines au moins après leur sortie nationale. Ceci permettrait de proposer plus de films, d'attirer plus de public et donc d'espérer un meilleur équilibre financier. Le coût élevé d'un système de projection (plus de 50 000 €) complique l'investissement dans cette nouvelle salle. C'est pour ce type de cas, à la frange de l'exploitation cinématographique classique, que nous souhaitons développer notre système de projection pour un coût total inférieur à 15 000 € : la machine servant de système de projection valant environ 2 000 €, la plus grande part du coût pourra être dédiée au projecteur.

Nous pensons que le système que nous proposons pourra être utilisé pour le prochain renouvellement du parc numérique des petites salles de cinéma. Il sera économiquement plus adapté aux salles ayant une largeur d'écran inférieure à six mètres.

La solution de projection proposée est une solution ouverte, fondée sur du logiciel libre et utilisant du matériel standard. Le système vlc4dcp que nous proposons se compose d'une machine dédiée à la lecture de DCP, d'un projecteur et d'une chaîne son. La machine dédiée à la lecture est présentée ci-dessous. Tout type de projecteur peut être connecté au système (idem pour la chaîne son).

Le système que nous proposons comporte des différences avec le système défini par le DCI. La table 2.3 présente les différences entre les deux systèmes.

Le format DCP s'est généralisé dans l'exploitation cinématographique. Ce format se positionne de plus en plus comme un standard, aux dépens des autres formats numériques comme le Blu-ray. Les films de répertoire sont également transcodés dans ce format, tout comme les documentaires qui, même s'ils ne sont pas initialement prévus pour le cinéma, sont transformés en DCP.

Les images d'un DCP sont compressées au format JPEG2000. Nous cherchons une solution permettant de décoder les fichiers JPEG2000 à une cadence d'au moins 24 images par seconde, au format 2K (images de taille 2048 × 1080, comportant trois composantes couleur et une profondeur de 12 bits).

Caractéristique	Système DCI	Système vlc4dcp
Taille d'image	2K/4K	2K
3D	Oui	Non
Profondeur de l'image	12 bits	12 bits
Son	WAV 24 bits	WAV 24 bits
Sous-titres	Oui	Oui
Chiffrement de DCP	Oui	Oui
Gestionnaire de sécurité (SM)	Oui	Non
Sécurité physique	Oui	Non
Traces sécurisées	Oui	Non
Tatouage du contenu	Oui	Non

TABLE 2.3 – Comparaison des systèmes de projection DCI et vlc4dcp.

Pour la lecture de films au format DCP, nous avons décidé a priori de limiter les capacités de notre système par rapport au système DCI. Nous ne prévoyons pas la lecture de DCP en 4K, car notre système est pensé pour les petites salles. En revanche, ceci n'empêche pas de lire un DCP au format 4K et d'effectuer sa projection en 2K (comme cela est prévu par le DCI). De plus, nous ne prenons pas en compte la lecture de films au format 3D. L'utilisation de la 3D se limite aux *blockbusters*, et n'est donc adaptée ni au cinéma d'Art et Essai, ni aux petites salles.

Nous avons décidé de ne pas implémenter toutes les contraintes de sécurité définies par le DCI. Les caractéristiques liées à la sécurité, qui sont décrites dans la table 2.3, seront détaillées dans le chapitre 5. Nous sommes ici dans le cadre d'un prototype qui se focalise sur la lecture de DCP, car il s'agit de la partie la plus complexe du système. De nombreuses copies numériques étant chiffrées, nous avons implémenté les algorithmes de chiffrement et de déchiffrement de DCP conformément à la DCSS.

Par la suite, il est possible que nous décidions de ne pas mettre en place toutes les exigences de sécurité réclamées, et donc que nous ne soyons pas conformes au standard DCI. Cela constitue un débat plus économique que technique. En effet, si nous mettions en place tous les niveaux de sécurité, le coût de notre système risquerait d'augmenter et donc de ne plus répondre à l'attente des petites salles, qui souhaitent limiter les dépenses. Il faudra donc se poser la question de savoir si les risques pris sur la sécurité de l'œuvre sont en adéquation avec le modèle économique de la distribution de l'œuvre, sachant que, dans le domaine des films d'Art et Essai, il est souvent plus important de parvenir à diffuser une œuvre que de la protéger.

Le CNC, au travers de la Commission Supérieure et Technique de l'image et du son (CST), exige que les salles de cinéma françaises soient équipées de matériel conforme à la DCSS. Il y a donc aussi un enjeu politique pour les petites exploitations. Faut-il acheter du matériel conforme à la DCSS, au risque de mettre en péril l'équilibre financier de la salle, ou bien utiliser un système moins coûteux et rendre la salle viable? Cette question sera d'autant plus prégnante lors du prochain renouvellement des projecteurs.

Le CNC et l'État ont beaucoup aidé à financer la transition vers le numérique, ce qui ne sera très probablement pas le cas lors du prochain renouvellement du matériel.

Nous avons choisi d'utiliser du matériel standard, pour rendre le plus facile possible l'installation et l'utilisation du système de projection. De plus, le matériel standard est moins coûteux que le matériel dédié.

L'enjeu est donc bien de pouvoir proposer une solution qui permette de lire et de projeter des DCP (chiffrés ou non chiffrés), même si cette solution n'est pas complètement conforme aux spécifications du DCI.

L'outil vlc4dcp, si l'on réduit le système à la machine lisant les DCP, permet également de visualiser des DCP sur un écran. Nous n'avions pas pensé à cette fonction au début du projet, mais nous avons pu constater qu'il y avait des demandes. En effet, il n'existe pas de solution libre et gratuite pour prévisualiser des DCP sur une machine standard. Il existe des outils pour générer des DCP, mais pas d'outil libre pour visualiser le résultat. Des projectionnistes réalisent des bandes-annonces qu'ils transforment en DCP, et n'ont pas d'autre solution pour voir le résultat que de charger le DCP dans un serveur de projection. Cela est plutôt contraignant, par exemple s'il y a une projection en cours. Avec notre système, il sera facile de répondre également à cette demande.

Nous présentons maintenant une configuration plus détaillée de vlc4dcp, qui est celle que nous avons développée durant cette étude dans le cinéma Utopia de Tournefeuille. Cette configuration est présentée sur la figure 2.9.

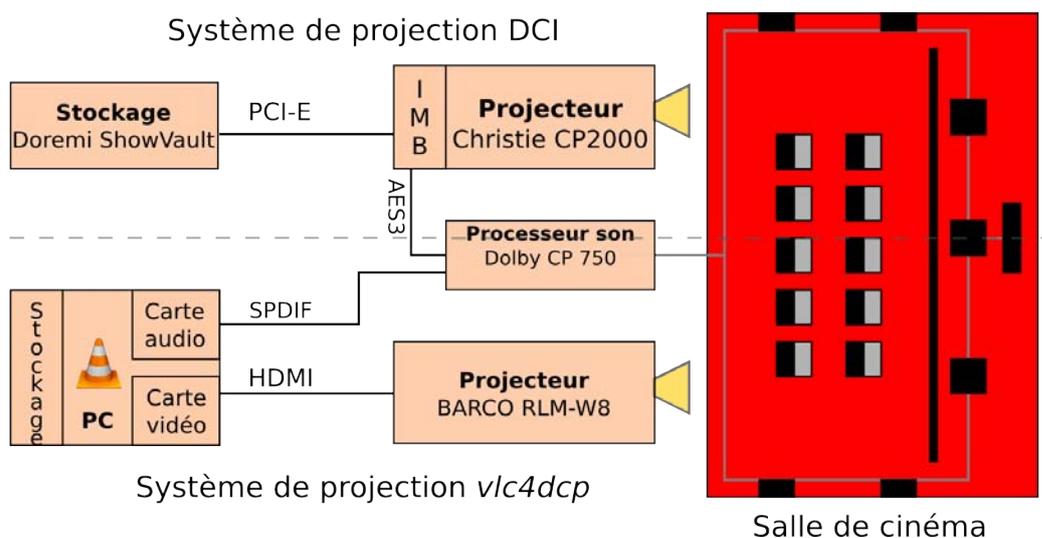


FIGURE 2.9 – Systèmes de projection pour le cinéma numérique. Dans le système DCI, le MB, intégré au projecteur, réalise le déchiffrement et la décompression. Dans notre système, il n'est pas intégré et peut donc être connecté à tout type de projecteur.

Du point de vue du matériel, les principales différences concernent la localisation du module de décodage et la connectique. Dans un système de projection DCI, la lecture du DCP est effectuée par la carte IMB incluse dans le projecteur. Dans le système

Caractéristique	Projecteur DCI	Projecteur alternatif
Définition	2048 × 1080	1920 × 1080
Profondeur	12 bits	10 bits
Débit supporté	3 Gb/s	3 Gb/s
Sous-échantillonnage	4 : 4 : 4 (aucun)	4 : 2 : 2
Coût	Plus de 50 000 €	De 10 000 € à 20 000 €
Largeur d'écran maximale	30 m	10 m

TABLE 2.4 – Comparaison des propriétés des projecteurs pour le cinéma numérique.

vlc4dcp, le déchiffrement et le décodage ne sont pas intégrés au projecteur. Ceci permet une plus grande modularité, mais s'oppose aux recommandations actuelles des *majors* américaines. Le MB intégré n'est pas une exigence du DCI, mais une demande des *majors*. Ce que le DCI exige, c'est qu'il y ait un lien sécurisé entre le MB et le projecteur. Pour cela, nous pouvons utiliser une interface *High Definition Multimedia Interface* (HDMI), qui supporte la définition du 2K et les 12 bits de profondeur de couleur¹⁰. L'interface HDMI peut également supporter un lien chiffré, au travers du protocole *High-bandwidth Digital Content Protection* (HDCP)¹¹.

Les connectiques pour le son ne sont pas identiques non plus. Nous utilisons une sortie *Sony/Philips Digital InterFace* (SPDIF) qui peut être connectée sur le même processeur son que le projecteur DCI. Le SPDIF est un connecteur audio optique et numérique, désormais très courant sur les cartes son.

Les constructeurs de l'industrie cinématographique proposent du matériel dédié mais, pour un écran de largeur inférieure à dix mètres, il est possible d'utiliser des projecteurs moins coûteux, fondés sur la même technologie que les projecteurs 2K. Des tests ont montré que cela permettait d'obtenir une bonne qualité d'image à la projection [4]. Ces deux types de projecteurs ont des propriétés très similaires (cf. table 2.4).

La définition de l'image est un peu différente : un projecteur DCI comporte 128 pixels de plus en largeur. Les films projetés sont rarement en pleine échelle 2K. Le rapport d'aspect d'un film, au moment du tournage, vaut souvent 2,39 ou 1,85, ce qui donne, respectivement, une résolution de 2048 × 857 ou de 1998 × 1080. Il ne faut donc que très légèrement modifier l'image pour être compatible avec le projecteur alternatif. Cette mise à l'échelle peut se faire pendant la décompression ou après.

Un projecteur alternatif a une profondeur de couleur de 10 bits, au lieu de 12 bits pour le projecteur DCI. Notre système étant dédié aux petits écrans, cette différence n'est pas perceptible. De plus, très peu de caméras numériques sont capables de prendre

10. Le HDMI est une norme et une interface numérique, pour transmettre des flux audio et vidéo non compressés. Ces flux peuvent être chiffrés. Pour pouvoir supporter le format du 2K, il faut que les interfaces soient compatibles avec au moins la version 1.3 du protocole. À partir de la version 1.3, la résolution maximale est 2560 × 1600 et supporte une profondeur de couleur de 16 bits.

11. Le HDCP est une méthode de protection numérique contre la copie de la vidéo et du son sur un connecteur multimédia (par exemple le connecteur HDMI). Il faut que la version du protocole HDCP soit au moins égale à la version 2.2 pour que le chiffrement soit sûr.

des images avec 12 bits de profondeur. De même, les caméras procèdent à un sous-échantillonnage chromatique (les pixels ne sont pas tous codés sur trois canaux), ce que proscrit pourtant le DCI.

Nous avons pu comparer aux projecteurs DCI le projecteur alternatif Barco RMLW8, qui est utilisé par le cinéma Utopia de Tournefeuille pour des salles de 150 places. Le prix de ce projecteur est un peu supérieur au prix annoncé pour notre système de projection, mais il s'agissait du seul projecteur disponible pour cette étude. Il existe des projecteurs ayant des caractéristiques similaires, certes un peu moins lumineux, mais à des tarifs allant de 10 000 € à 15 000 €.

Le décodage et le déchiffrement se font sur un PC standard. Les caractéristiques du type de matériel utilisé sont les suivantes :

Processeur – Intel core i7 possédant au moins 4 cœurs. Le traitement le plus complexe est le décodage JPEG2000. Il est réalisé en parallèle par ces processeurs. Le traitement est présenté dans le chapitre 3 pour la partie concernant la gestion de DCP avec VLC, et dans le chapitre 4 pour celle qui concerne le format JPEG2000.

Mémoire – 8 ou 16 Gb de RAM. Il faut une quantité de RAM plutôt élevée pour le décodage JPEG2000 d'images au format 2K. Une image décompressée peut avoir une taille de 12 Mo et est stockée en RAM à la fin de la décompression. On privilégie l'utilisation de disques *Solid-State Drive* (SSD) pour la lecture, qui sont bien plus rapides que les disques rotationnels classiques.

Stockage – Un disque SSD de petite taille pour le système d'exploitation GNU/Linux, un SSD de 500 Go pour la lecture de DCP, et un ou plusieurs disques rotationnels pour le stockage de DCP.

Vidéo – Une carte vidéo de milieu de gamme, avec sortie HDMI, une autre sortie vidéo pour le contrôle de la projection. Il faut néanmoins choisir une carte graphique avec environ 1000 cœurs graphiques (comme les cartes à base de NVIDIA GTX 760 ou RADEON R7 370) et 2 Gb de mémoire, afin de permettre des traitements de filtrage après la décompression, comme par exemple les changements d'espace colorimétrique qui se font de manière matricielle sur l'image.

Audio – Une carte audio supportant au moins la configuration de canaux 5.1 avec sortie numérique SPDIF.

Réseau – Deux cartes réseau : une pour se connecter au réseau de la cabine numérique ; une autre pour assurer la réception de contenu.

Le coût de cette machine, une fois montée, s'élève à 2 000 € environ.

2.5.2 Distribution de contenu dédiée

La DCSS [16] spécifie la création, la distribution, la projection et la sécurisation d'une copie numérique. Le document décrit en détail les formats de fichiers et le système de projection. À plusieurs reprises, l'interopérabilité est mentionnée mais néanmoins, le transport des copies numériques n'est pas détaillé (une seule page sur 155). Le transport peut être physique (envoi de films au format DCP sur disque dur) ou bien dématérialisé

(i.e. par les réseaux). Ce manque de spécification a eu pour conséquence l'émergence d'un faible nombre d'offres de transport dématérialisé. Distributeurs et cinémas se trouvent aujourd'hui face au besoin de s'équiper de toutes les offres pour assurer la distribution des films. Les coûts de transfert dématérialisé étant élevés, tous les distributeurs, notamment les plus petits et les plus fragiles, ne peuvent pas s'équiper et une bonne partie d'entre eux continuent à envoyer les DCP par voie postale. De plus, nous assistons à une concentration des opérateurs de l'industrie cinématographique (rapprochement Ymagis/Smartjog et Globecast/Eclair) à l'échelle européenne. Cette situation est dangereuse pour la diversité de l'exploitation et de la distribution. Or, Internet étant un réseau neutre, il est tout à fait possible de mettre en œuvre une solution indépendante avec des technologies déjà existantes et libres.

Un distributeur négocie les droits d'exploitation d'une œuvre, généralement par pays, avec un réalisateur/producteur. Ensuite, il négocie avec les exploitants la projection du film en salle. La relation producteur/distributeur/exploitant est surtout d'ordre contractuel. La distribution physique du film est faite, en pratique, par des sociétés spécialisées.

Le DCP chiffré est envoyé aux salles via un transporteur de contenu. Ce transporteur peut être physique (le DCP chiffré est copié sur un disque dur et expédié par courrier postal aux exploitants). À l'origine du cinéma numérique (2009-2012), cela constituait le principal moyen de transport. La même logistique de circulation que les bobines (circulation de copies de salle en salle) était alors utilisée. Sont arrivées ensuite des propositions de transport dématérialisé par le réseau Internet et par satellite, avec des opérateurs comme Globecast ou Smartjog. Le but est d'effectuer le transfert depuis le laboratoire du distributeur vers plusieurs exploitants, souvent en même temps (sorties nationales, par exemple pour *Bilbo le Hobbit* : 900 copies en France).

Cela pose alors le problème suivant : si un exploitant souhaite pouvoir programmer des films provenant de n'importe quel distributeur, doit-il s'équiper de tous les moyens de transport dématérialisé ? Inversement, si un distributeur souhaite acheminer son film vers n'importe quelle salle, le laboratoire doit-il être équipé de tous les moyens de transport ? Se pose donc le problème de l'interopérabilité entre laboratoires et exploitants, le principal problème étant le coût pour s'équiper de plusieurs offres.

Le cas de Globecast

Les offres commerciales ne sont pas toutes les mêmes : elles dépendent du type d'exploitation. Prenons le cas de Globecast, qui équipe gratuitement certaines salles, mais pas d'autres. Le critère de cette gratuité n'est pas connu, mais on constate que cela concerne les exploitations les plus grandes ou ayant une grande diversité de programmation (salles dites « stratégiques »), les multiplexes, mais aussi des cinémas comme l'Utopia de Bordeaux (cinq salles, beaucoup de sorties nationales Art et Essai). La gratuité prend en compte l'abonnement à deux ou quatre lignes ADSL (avec agrégation de lignes, en cas d'absence de fibre optique), le matériel de réception fourni, et le transfert illimité de DCP. Dans ce type d'offre, c'est le distributeur qui prend en charge tous les coûts via la facturation de chaque transfert des *data center* de Globecast vers les salles. Les coûts pour le distributeur, pour une sortie nationale, sont de 50 € par transfert et

par salle. Pour une projection du film quelques semaines après la sortie nationale, le coût varie de 20 € à 30 € par transfert et par salle. Il y a aussi une facturation pour le premier chargement du film sur le *data center* de Globecast. Cette tarification est rebutante pour les plus petits distributeurs. Ainsi, même si l'Utopia de Bordeaux est équipé gratuitement, seuls 8 % des films projetés sont reçus par Globecast. Le cinéma reçoit le reste des films par disques durs.

Pour les petites et moyennes salles, Globecast propose une offre payante, avec 3 000 € de frais de mise en service, puis des abonnements variant de 200 € à 300 € par mois, selon le nombre de films à envoyer. Même si une exploitation choisit cette offre, elle n'accède pas à tous les films, car Globecast ne travaille pas avec tous les distributeurs.

Alors, que faire ? Travailler avec tous les transporteurs de contenu : c'est ce que font certaines salles. Cela implique non seulement un coût élevé, mais également une complexité technique non négligeable : nombreuses lignes téléphoniques, satellites, équipements hétérogènes, et donc maintenance plus compliquée.

Les transporteurs de contenu n'étant pas interconnectés, chacun fournit sa solution jusqu'au cinéma, même si les opérateurs proposent des connexions séparées et qu'elles transitent pour finir par les mêmes « tuyaux ». Pour faire un parallèle avec les autoroutes, tout se passe comme si les différents usagers utilisaient la même autoroute mais passaient, à la sortie, par un péage dépendant de l'opérateur choisi.

Un système de transport ouvert et interopérable

Le principe fondamental de notre solution est de proposer un système de distribution de DCP interopérable, qui soit économiquement viable pour les distributeurs et les exploitants. Par « interopérable », nous entendons un standard public pour le transfert de DCP basé sur des logiciels libres gratuits et indépendants des fournisseurs d'accès à Internet, des distributeurs, et des intégrateurs de système de projection.

Dans l'exploitation cinématographique, les solutions proposées par Globecast sont adaptées à la distribution massive de DCP pour les grandes exploitations (multiplexes).

Notre solution s'adresse à la petite et moyenne exploitation d'Art et Essai, qui projette beaucoup de films. En France, en 2013, 60 % des films projetés l'ont été dans moins de 100 salles (sources CNC) et la grande majorité de ces films est classée Art et Essai.

De plus, les transporteurs dématérialisés de DCP cherchent à obtenir des contrats d'exclusivité avec les grands distributeurs. Ceci ne facilite pas la réception de DCP par tous les exploitants et s'oppose au principe d'interopérabilité de la DCSS.

Le transport physique de DCP est un protocole interopérable et économique, qui ne dépend ni d'un distributeur, ni d'un transporteur, ni du matériel de projection. C'est un moyen de transport économique. Son défaut majeur est qu'il faut gérer et suivre la circulation des DCP. Quand le nombre de salles à alimenter est élevé, cela peut devenir complexe. C'est une des raisons pour lesquelles les grands distributeurs souhaitent passer au transport dématérialisé. Mais pour des films avec des délais de distribution restreints, le transport physique reste une solution avantageuse.

Le principal problème du transport dématérialisé est la taille des DCP, qui peut atteindre 400 Go pour un film de 3 heures compressé à un débit binaire de 250 Mb/s.

La taille moyenne observée d'un DCP est de 200 Go.

Le transport dématérialisé pour le cinéma d'Art et Essai doit pouvoir remplacer avantageusement le transport physique. L'avantage du transport dématérialisé et que l'on peut expédier le même contenu simultanément vers plusieurs cinémas. Il faut aussi que le temps de transfert n'excède pas le temps de transport physique : nous nous sommes fixé une durée maximale de 48 heures.

Proposer un système interopérable et adapté à l'économie de la distribution de films d'Art et Essai est à l'origine du projet The DCP Bay (référence au célèbre traqueur BitTorrent suédois « The Pirate Bay »). Ce projet regroupe plusieurs partenaires : les cinémas Utopia, le GNCR et des Fournisseur d'Accès à Internet (FAI) associatifs, au travers de la Fédération FDN (FFDN). Tous ont un intérêt commun à promouvoir un système de distribution permettant le transport de toute œuvre, quelque soit son audience. Ce projet constitue la deuxième contribution de cette thèse, en réponse aux problèmes de distribution des films précédemment identifiés.

Les cinémas Utopia ont financé cette étude. Le GNCR nous a apporté son réseau de contact avec les distributeurs, ainsi qu'une interface « commerciale ». La FFDN nous a proposé d'héberger dans ses infrastructures les serveurs nécessaires à la distribution, et de nous faire profiter de ses connexions réseau et de son savoir-faire en matière de « déploiement réseau ». Le système envisagé doit permettre de distribuer environ 1000 DCP par mois, vers plus de 200 établissements, pour être économiquement viable, afin d'assurer l'activité de deux salariés et de financer les infrastructures réseau.

Le système The DCP Bay se fonde sur le modèle de distribution « pair à pair » (*peer-to-peer*), et notamment le protocole BitTorrent. Ce protocole, très utilisé sur Internet, est le mieux adapté au transport de fichiers volumineux car il est souple, robuste et sa mise en œuvre est simple. Il intègre « nativement » un contrôle d'intégrité des données reçues. Il répond également au besoin d'interopérabilité : sa spécification est librement accessible, et il existe de nombreuses implémentations en logiciel libre.

La sécurité du transfert des copies numériques est garantie par l'utilisation de *Virtual Private Network* (VPN). Il existe de nombreuses solutions de VPN. Nous avons deux contraintes majeures à propos du VPN : une implémentation libre, et qui puisse supporter un fonctionnement pair à pair. Cela réduit fortement les choix. Nous avons choisi le VPN Tinc, qui répond à ces deux contraintes, et dont la mise en place est simple.

Pour le stockage des DCP et la gestion de la bande passante pour le transfert, nous nous appuyons sur des FAI associatifs, notamment les membres de la FFDN.

Une fois réceptionnés (par disques durs ou par FTP), les films au format DCP sont stockés sur des serveurs en *data center*. Leur intégrité est vérifiée, puis un fichier Torrent est créé pour initier les transferts. Plusieurs serveurs, appelés *seeders*, se synchronisent pour assurer une bonne distribution initiale. Un exploitant peut alors commencer le téléchargement du DCP à partir du fichier Torrent reçu et du client BitTorrent.

Le système proposé est donc très simple à utiliser. Il ne nécessite pas d'investissement coûteux en matériel pour l'exploitant, et s'appuie sur des accès à Internet classiques (ADSL, VDSL). De plus, l'exploitant peut participer plus efficacement à la diffusion du film vers d'autres salles, s'il est équipé d'un accès à très haut débit de type fibre optique.

La plus grande partie d'un DCP est composée des fichiers JPEG2000. Nous avons exclu les solutions de transfert utilisant les propriétés du format JPEG2000, comme les différents niveaux de définition [46] et les systèmes de distribution de type *multiple-description-based* [8]. En effet, les DCP étant généralement chiffrés, nous ne pouvons pas accéder directement aux données JPEG2000.

2.6 Bilan

Nous avons présenté le contexte de la projection numérique dans l'exploitation cinématographique, de la distribution à la projection en salle. En 2015, 100 % des salles sont équipées de matériel de projection numérique.

Les salles ont été équipées grâce à divers mécanismes d'aides financières. Ces aides ne seront probablement pas reconduites lors du prochain renouvellement de matériel. Le matériel de projection est coûteux, surtout pour les petites salles. Les solutions de projection actuelles sont matérielles et « propriétaires ». Nous avons présenté un système de projection alternatif, la lecture de DCP se faisant avec VLC : le détail de l'implémentation, ainsi que des résultats, seront présentés dans le chapitre 3. Une des principales difficultés rencontrées a été le décodage du format JPEG2000. Dans le chapitre 4, une solution de décodage logiciel fondée sur la bibliothèque libavcodec sera détaillée.

La sécurité est prépondérante dans la DCSS [16] : le chapitre 5 traitera de la création de DCP chiffrés, puis de leur déchiffrement dans VLC.

Pour la distribution de contenu DCP, nous avons mis en place une solution opérationnelle, dont le principe et la mise en œuvre seront détaillés dans le chapitre 6.

Chapitre 3

Contributions au lecteur multimédia VLC

3.1 Introduction

Notre objectif est de pouvoir lire un DCP à l'aide de logiciels libres. Même si nous savons que la principale difficulté est de décoder rapidement (à raison de 24 images par seconde au moins) les images JPEG2000, il faut également développer une interface pour le rendu et la visualisation des DCP. Cette interface doit être capable d'analyser le contenu d'un DCP, d'identifier les médias et de synchroniser leur lecture : il faut que les images, l'audio et, s'ils existent, les sous-titres soient lus de manière coordonnée.

Plutôt que de développer notre propre interface de visualisation, nous avons préféré utiliser des logiciels existants. Le choix de VLC s'est fait assez naturellement. Il s'agit d'un logiciel libre de lecture de vidéos très utilisé, multi-plateforme et doté d'une communauté de développeurs très active.

3.2 Choix d'un lecteur multimédia

Notre premier critère de sélection est le choix de logiciels libres pouvant être exécutés sous GNU/Linux. La table 3.1 compare les principaux lecteurs multimédia respectant

Nom	Dernière version	Moteur/ <i>framework</i>	Commentaire
Kaffeine	Avril 2011	gstreamer, xine-lib, ffmpeg	Peu maintenu
Mplayer	Mai 2013	ffmpeg	Maintenu, actif
Totem	Septembre 2014	gstreamer	Maintenu
VLC	Avril 2015	Interne	Maintenu, actif
Xine	Février 2012	Interne (xine-lib)	Peu maintenu

TABLE 3.1 – Comparaison de lecteurs multimédia libres sous GNU/Linux.

cette contrainte. Nous avons laissé de côté les applications plutôt dédiées à la gestion et à l'écoute de musique. Nous avons sélectionné les lecteurs multimédia capables de gérer tous types de flux vidéo en entrée. Avec ces critères, la liste se limite à cinq lecteurs. Kaffeine et Xine ne sont pas dotés de communautés de développeurs très actives. Totem est plutôt conçu pour être intégré dans l'environnement de bureau Gnome. Il repose sur le *framework* gstreamer, qui est un projet indépendant. Nous aurions pu écrire le décodeur JPEG2000 pour gstreamer, mais nous avons préféré le développer pour ffmpeg (et libav), car cette bibliothèque est nettement plus utilisée. Il restait donc à choisir entre Mplayer et VLC : c'est à nouveau l'organisation des communautés de développeurs qui a guidé notre choix. En fait, nous avons tout simplement eu plus de répondant de la part de VLC, lors des premiers contacts sur l'écriture du module VLC. De plus, VLC est un logiciel multiplateforme. Même si nous nous sommes centrés sur le développement sur GNU/Linux, le portage vers les autres systèmes d'exploitation sera plus aisé, d'autant plus que le logiciel VLC compile ou lie toutes les bibliothèques nécessaires à son exécution, quel que soit l'OS supporté.

3.3 Présentation de VLC

Le logiciel VLC est un logiciel libre de lecture multimédia diffusé à très grande échelle, téléchargé plus de 1,5 milliard de fois. Il est disponible pour de nombreux systèmes d'exploitation comme Linux, Windows, MacOS, mais est également distribué pour les OS des *smartphones* comme iOS ou Android.

3.3.1 Architecture de VLC

VLC est conçu comme un *framework* (système) multimédia qui charge dynamiquement des modules selon les entrées (fichiers, flux réseau) et les sorties (audio et/ou vidéo, sur écran ou sur réseau).

L'architecture de VLC est divisée en deux parties principales : le noyau et les modules. Le noyau est chargé des opérations de bas niveau comme le *threading*, l'horodatage et la synchronisation. Il fournit aussi les « tampons de flux » (pipelines) nécessaires à la communication entre modules.

Il existe plus de 400 modules dans VLC, qui sont chargés durant l'exécution suivant les besoins. Les modules sont catégorisés en types, selon leur fonction. Nous présentons ici les principaux types de modules :

access – Modules qui gèrent les entrées (fichiers, flux réseau, caméras, ...).

demux – Modules qui extraient les données en les séparant en flux audio, vidéo, ...
C'est ici que les conteneurs sont gérés. Exemple de demux : mkv.

access_demux – Modules qui sont à la fois access et demux. Ces modules sont utilisés dans le cas où l'on n'a pas une séparation directe entre l'accès aux données et le contenu, comme par exemple pour la lecture de DVD ou de Blu-ray.

decode – Ici se trouvent les modules chargés du décodage des données.

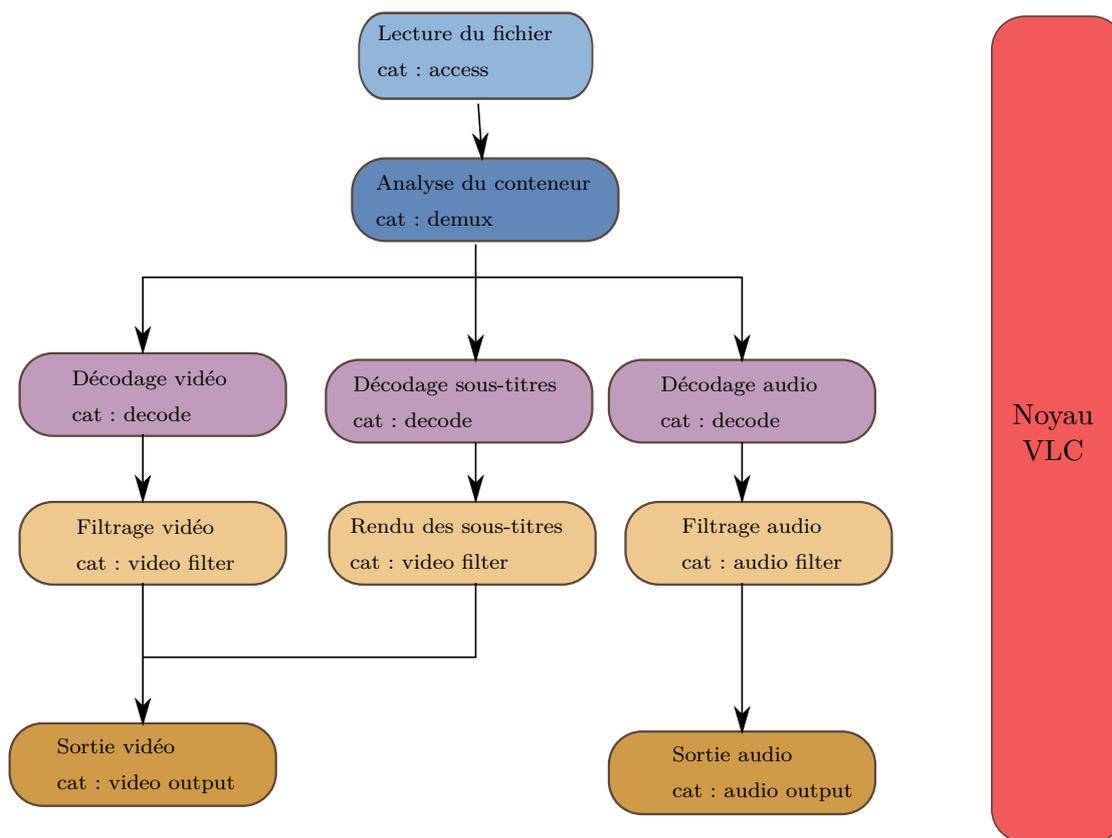


FIGURE 3.1 – Architecture du logiciel VLC (« cat » signifie catégorie). À gauche, représentation des modules VLC utilisés lors de la lecture locale d’un fichier vidéo. À droite, le noyau VLC chargé des opérations de bas niveau et de la communication entre modules.

video_output – Modules qui gèrent les sorties vidéo et l’interface avec l’environnement graphique de l’OS (exemples : OpenGL, framebuffer, ...).

audio_output Modules gérant les sorties avec les périphériques audio (exemples : ALSA, OSS, ...).

video_filter et audio_filter – Modules gérant des filtres pour la vidéo et l’audio.

Le séquençement de l’utilisation des modules pour lire un fichier vidéo avec sous-titres est présenté sur la figure 3.1.

Les modules sont souvent des implémentations d’interfaces (dans le sens API) vers d’autres bibliothèques. VLC dépend de plusieurs centaines de bibliothèques, dont libavcodec (implémenté dans libav et ffmpeg). On trouve aussi plusieurs modules remplissant les mêmes fonctions, mais avec de légères différences comme, par exemple, la dépendance à des bibliothèques différentes. Par exemple, pour la lecture de VCD (vidéo CD), il existe deux modules de type access. VLC gère le choix du module par un mécanisme de priorité.

Il essaie d'abord de traiter le contenu avec le module de plus forte priorité. Si cela échoue, il tente avec le module de plus faible priorité. Les conteneurs et les codecs ne sont pas toujours implémentés exactement de la même manière dans les différentes bibliothèques. Ce mécanisme de priorité par module permet ainsi de tester plusieurs méthodes sur le même contenu.

Chaque module chargé par VLC est exécuté dans un *thread* différent. Les modules communiquent entre eux au travers des pipelines fournis par le noyau de VLC.

VLC ne charge pas entièrement un contenu, même lorsqu'il est stocké localement, avant de le lire. Il traite continuellement le flux de données, un *thread* s'occupant de la lecture du contenu, un autre du décodage, un autre des sorties. Tout cela est synchronisé au travers d'un mécanisme d'horodatage appelé *Presentation Time Stamp* (PTS).

3.3.2 Écriture d'un module dans VLC

L'écriture d'un module dans VLC se fait à partir d'un canevas de module. Il faut commencer par déclarer le type du module et sa fonction. Il faut ensuite écrire des fonctions `Open`, `Close` et `Control` puis, en fonction du rôle du module, une fonction `Demux` pour un *demuxer* ou `Decode` pour un décodeur.

Listing 3.1 – Exemple de module extrait du site *VLC Developers Corner* (https://wiki.videolan.org/Hacker_Guide/How_To_Write_a_Module)

```

1
2 /* Module descriptor */
3 vlc_module_begin()
4     set_shortcode(N_("Hello"))
5     set_description(N_("Hello_interface"))
6     set_capability("interface", 0)
7     set_callbacks(Open, Close)
8     set_category(CAT_INTERFACE)
9     add_string("hello-who", false)
10 vlc_module_end()
11
12 /* Internal state for an instance of the module */
13 struct intf_sys_t
14 {
15     char *who;
16 };
17
18 static int Open(vlc_object_t *obj)
19 {
20     intf_thread_t *intf = (intf_thread_t *)obj;
21
22     /* Allocate internal state */
23     intf_sys_t *sys = malloc(sizeof(*sys));
24     if (unlikely(sys == NULL))
25         return VLC_ENOMEM;
26     intf->p_sys = sys;

```

```

27     /* Read settings */
28     char *who = var_InheritString(intf, "hello -who");
29     if (who == NULL)
30     {
31         msg_Err(intf, "Nobody to say hello to!");
32         goto error;
33     }
34     sys->who = who;
35     msg_Info(intf, "Hello %s!", who);
36     return VLC_SUCCESS;
37 }
38
39 static void Close(vlc_object_t *obj)
40 {
41     intf_thread_t *intf = (intf_thread_t *)obj;
42     intf_sys_t *sys = intf->p_sys;
43     msg_Info(intf, "Goodbye %s!");
44     /* Free internal state */
45     free(sys->who);
46     free(sys);
47 }

```

Le listing 3.1 montre un exemple de module. Dans cet exemple, un module de type interface ne fait qu'afficher le classique Hello! à l'ouverture et Goodbye! à la fermeture. Les lignes 2 à 10 correspondent à la déclaration du module dans VLC : VLC traitera ces lignes pour savoir quel type de module est déclaré. Ici, on a un module de type interface, dont la priorité est égale à 0 (`set_capability("interface",0)`), ce qui signifie que l'utilisateur doit explicitement appeler ce module au démarrage pour le charger). Le module s'appelle Hello (`set_shortcode(N_("Hello"))`).

Tout module doit avoir des fonctions Open et Close. Le noms de ces fonctions sont donnés au noyau VLC via la directive `set_callbacks(Open,Close)`. On retrouve le corps des fonctions Open et Close dans le listing. La première fonction affiche Hello!, la deuxième Goodbye!. La variable `who` est une variable du module. Elle peut être paramétrée à l'extérieur du module, par exemple lors du lancement de VLC.

3.3.3 Organisation du code dans VLC

Le code source de VLC est librement disponible. Il est géré par le logiciel de gestion de version git. On peut donc facilement le récupérer grâce à la commande :

```
1 git clone git://git.videolan.org/vlc.git
```

La figure 3.2 montre l'organisation du code. Le code est réparti entre noyau (répertoires `include` et `src`) et modules (répertoire `modules` et sous-répertoires organisés par catégories). Tous les répertoires du code source VLC n'ont pas été représentés, notamment ceux qui servent pour la compilation de VLC, la traduction, ... De même, tous les sous-répertoires de modules n'ont pas été représentés.

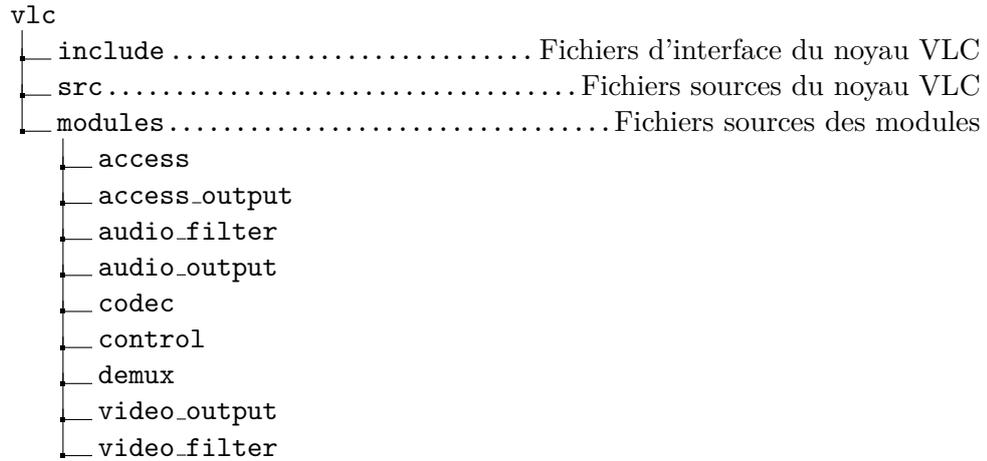


FIGURE 3.2 – Organisation du code source de VLC. Pour améliorer la lisibilité, tous les répertoires inclus dans le code source de VLC n'ont pas été représentés.

On constate donc que, pour créer un module dans VLC, il faut stocker ses sources dans le sous-répertoire correspondant à la catégorie du module. Le code source peut être écrit en C ou en C++.

3.4 Module DCP de VLC

Nous présentons dans ce paragraphe notre contribution au lecteur multimédia VLC pour la lecture de DCP.

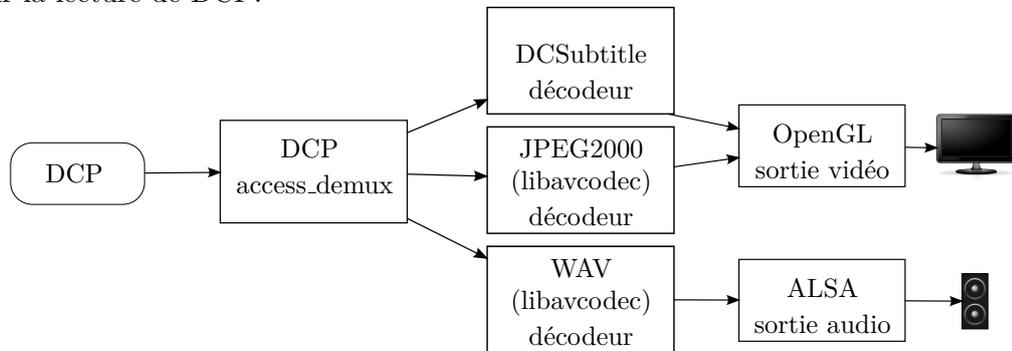


FIGURE 3.3 – Lecture de DCP avec VLC et libavcodec. Chaque boîte représente un module VLC. Les flèches représentent le flux des médias entre les différents modules.

La figure 3.3 résume la lecture de DCP avec VLC. Le DCP arrive en entrée sous la forme d'un répertoire de disque dur. Le module DCP démultiplxe le DCP et fournit aux décodeurs les flux de données. Une fois décodés, les flux sont envoyés vers les périphériques de sortie vidéo et audio.

Dans VLC, il n'existait aucun module implémenté pour la lecture de DCP. Notre première contribution a été de créer un module pour gérer les DCP. Nous nous focalisons ici sur la gestion d'un DCP, le décodage JPEG2000 traité par la bibliothèque libavcodec étant décrit dans le chapitre 4.

3.4.1 Structure d'un DCP

Un DCP est un répertoire (cf. figure 2.6) contenant des fichiers de méta-données au format XML, et des fichiers de contenu multimédia stockés au format MXF, sauf pour les sous-titres qui peuvent être stockés soit au format XML, soit au format PNG (dans ce cas, il s'agit d'une suite d'images).

Nous avons initialement choisi d'écrire un module VLC de type access, et ensuite de laisser le module libavformat traiter le démultiplexage. Cela n'était pas un bon choix. En effet, un module demux attend en entrée un seul flux, ce qui n'est pas notre cas car nous avons plusieurs flux pour la vidéo, le son et les sous-titres. Nous aurions pu contourner ce problème en créant plusieurs flux pour chaque type de données et en lançant un module demux pour chaque flux. Mais cela aurait été compliqué dans le cas d'un film constitué de plusieurs bobines virtuelles. De plus, libavformat n'utilisant pas ASDCPlib, il aurait aussi fallu dans ce cas créer un module demux pour l'utilisation de ASDCPlib.

Il faut voir un DCP comme un conteneur (le répertoire du DCP) contenant d'autres conteneurs (les fichiers MXF). Et de plus, dans le cas où le film est distribué en plusieurs langues, le DCP peut contenir plusieurs fichiers MXF pour l'audio et plusieurs fichiers pour les sous-titres.

3.4.2 Choix du type de module

Cette organisation des contenus est spécifique au DCP : c'est ce qui nous a poussés à choisir un module de type access_demux. D'autres formats ont également comme point d'entrée un répertoire, comme les DVD ou les Blu-ray, mais nous ne sommes pas dans le même cas. Le contenu multimédia d'un DVD est découpé en plusieurs fichiers *Video Object* (VOB), mais le découpage est temporel, un fichier VOB étant un conteneur pour le MPEG2 dans lequel on retrouve les données compressées pour l'audio et la vidéo.

Nous avons donc choisi de créer un module de type access_demux. Ce type de module permet d'avoir plus de souplesse dans l'organisation du module DCP. Il prend en entrée un chemin vers un DCP et fournit en sortie des flux de type *Elementary Stream* (ES). Dans VLC, un ES est un flux vidéo, audio ou de sous-titres qui a été démultiplexé par le module gérant les conteneurs et est prêt à être traité par les décodeurs. Les entrées/sorties du module sont représentées sur la figure 3.4.

Les en-têtes des DCP étant écrits en XML, nous utilisons un module VLC dédié pour analyser les fichiers XML et identifier les médias.

3.4.3 Bibliothèque ASDCPlib

La bibliothèque ASDCPlib a été lancée par le DCI. Elle est maintenant développée par l'entreprise CineCert. C'est une bibliothèque libre qui permet de lire et écrire des fichiers au format MXF. Elle est très utilisée dans l'industrie cinématographique pour la création de DCP.

Elle implémente le traitement du conteneur MXF, et en particulier celui du cinéma numérique défini par la SMPTE. Elle permet aussi de gérer des contenus chiffrés.

Nous avons choisi d'utiliser cette bibliothèque dans notre lecteur de DCP. Il existe des implémentations du conteneur MXF dans libavformat, mais qui ne sont pas aussi abouties que ASDCPlib. Dans le conteneur MXF de libavformat, on ne retrouve pas l'implémentation de toutes les spécifications de la SMPTE.

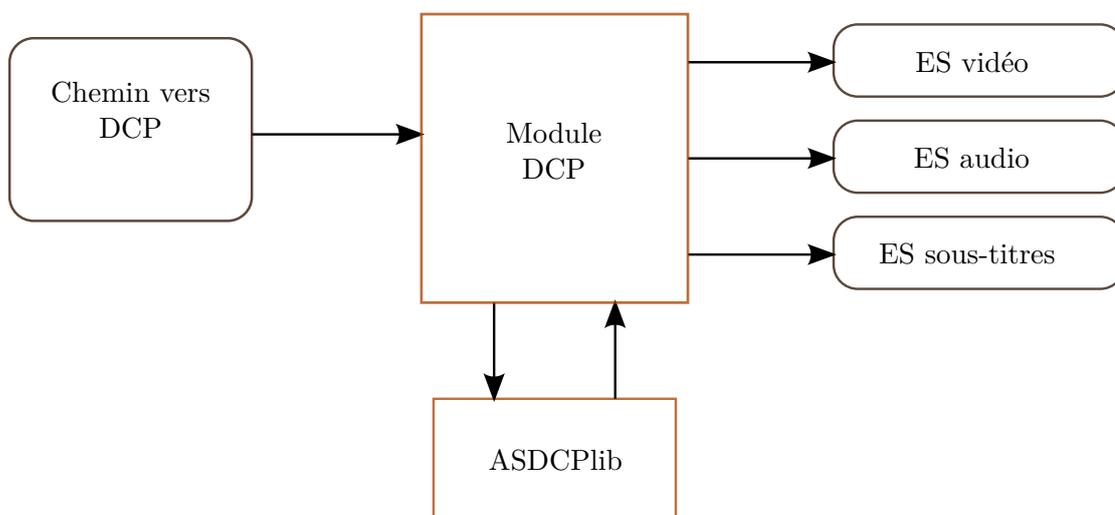


FIGURE 3.4 – Entrées/sorties du module DCP. Le module reçoit le chemin où est stocké le DCP en entrée. En sortie, il produit un ES pour chaque type de contenu. La lecture des fichiers MXF est faite via la bibliothèque ASDCPlib.

On lance la lecture d'un DCP dans VLC grâce à la ligne de commande suivante :

```
1 vlc dcp:///chemin/vers/un/dcp
```

L'indication `dcp://` demande à VLC d'aller explicitement charger notre module au démarrage.

3.4.4 Implémentation du module DCP

L'algorithme 1 décrit l'initialisation du module DCP. À la ligne 2, les structures de données sont créées et initialisées : cela représente toutes les variables qui seront utilisées par le module.

Algorithme 1 Initialisation du module DCP.

```
1: function OPEN
2:   Initialisation des structures de données du module DCP.
3:   Parcours des fichiers XML du DCP pour identification du contenu.
4:   Ouverture des flux de lecture MXF avec ASDCPlib.
5:   Création des ES, suivant le type de chaque donnée.
6: end function
```

La ligne 3 fait l'analyse des fichiers XML du DCP. Nous relevons toutes les informations nécessaires à la lecture du DCP, comme les noms des fichiers MXF, leurs durées, leur séquençement, ...

La ligne 4 initialise l'interface avec ASDCPlib. Pour chaque fichier MXF, on initialise les pointeurs vers les données à lire. Notons ici que l'on travaille toujours par zone de données partagées entre ASDCPlib et le module DCP : il faut éviter de faire des copies de données en mémoire. Cela ralentirait trop le parcours des fichiers MXF.

La ligne 5 initialise les ES à partir des données lues sur les fichiers MXF. C'est là, par exemple, que l'on va transmettre à VLC la taille des images ou la fréquence de lecture.

La fonction `Open` n'est appelée qu'une fois : c'est la fonction `Demux` qui sera appelée pour lire chaque trame du fichier MXF.

Algorithme 2 Fonction Demux du module DCP.

```
1: function DEMUX
2:    $frame\_no \leftarrow frame\_no + 1$ 
3:    $ES\_video \leftarrow \text{LIREFRAMEVIDEO}(frame\_no)$ 
4:    $ES\_audio \leftarrow \text{LIREFRAMEAUDIO}(frame\_no)$ 
5:   TRANSMETTRE( $ES\_video$ )
6:   TRANSMETTRE( $ES\_audio$ )
7: end function
```

La fonction `Demux` décrite dans l'algorithme 2 est appelée à chaque nouvelle trame demux de type vidéo à traiter. Nous avons pris la trame vidéo comme référence et comme PTS. La trame audio est calculée en fonction de la trame vidéo, de même pour les sous-titres que nous n'avons pas représentés ici.

Le traitement consiste à demander à la bibliothèque ASDCPlib de nouvelles trames (*frames*), qui sont stockées dans des zones précédemment allouées.

À chaque nouvelle trame, on calcule le PTS associé. VLC traite les différents modules dans des *threads* séparés, donc la fonction `Demux` est appelée de manière asynchrone par rapport au décodage. Par exemple, c'est le PTS associé à la trame qui assure la synchronisation. La transmission des ES entre le *demuxer* et le décodeur se fait par une FIFO gérée par le noyau de VLC.

Parmi les principales fonctions de ce module, on trouve la fonction `Control`, qui permet la communication entre le module et l'application. C'est là que l'on gère les interactions avec l'utilisateur, comme la mise en pause du film ou la navigation dans le contenu.

La fonction `Close` permet de terminer le traitement et de libérer l'espace mémoire.

Pour gérer le traitement des bobines virtuelles, on parcourt tous les fichiers de contenu pour pouvoir réaliser un index des trames. Ensuite, dans la fonction `Demux`, en fonction de la trame à charger, on modifie les fichiers MXF sur lesquels on fait l'acquisition. Vu que le traitement est asynchrone, on ne perturbe pas le traitement des décodeurs par le chargement de nouveaux fichiers MXF. On remplace plusieurs fichiers MXF à la fois car il faut charger les pistes vidéo, audio et sous-titres en même temps.

3.4.5 Décodage des contenus

Nous n'avons pas développé de module spécifique pour la vidéo et l'audio. Le traitement des sous-titres est un cas à part que nous verrons dans le prochain paragraphe.

Le décodeur JPEG2000 est dans `libavcodec`, le module gérant `libavcodec` dans VLC étant `avcodec`. Dans sa fonction `Open`, le module teste sur les premières trames s'il peut décoder du JPEG2000. Dans notre cas, il retourne OK, et VLC choisit ce décodeur.

Le son d'un DCP est encodé au format WAV, donc nous n'avons pu utiliser l'existant : c'est à nouveau `libavcodec` qui a été choisi pour décoder le son. Il a seulement fallu réordonner les pistes, notamment pour le son 5.1 et 7.1, pour une restitution spatiale correcte du son. En effet, l'ordre des pistes défini par la DCSS n'est pas le même que l'ordre défini dans les sorties son de VLC.

3.4.6 Rendu des sous-titres

Les sous-titres dans VLC sont traités comme un décodeur. Il faut fournir en entrée du décodeur le message (au format texte) à afficher. Le PTS associé au message permettra de l'afficher au bon moment. Le module d'affichage vidéo est chargé d'afficher le texte au-dessus du flux vidéo décodé.

La DCSS définit un format de sous-titres en XML qui n'était pas encore implémenté dans VLC. Un module de type décodeur, dédié au décodage de ces sous-titres, doit être développé. Ce format de sous-titres pourrait ainsi être utilisé par d'autres formats que le DCP. Le format de sous-titres défini par la SMPTE [44] permet, en plus du rendu du texte synchronisé, de préciser où le sous-titre doit être affiché dans l'image, en quelle couleur et avec quelle police (la police à utiliser est fournie avec le DCP).

Les sous-titres d'un DCP peuvent également être fournis sous forme d'images au format PNG à afficher au-dessus de la vidéo. Dans ce cas également, un décodeur de sous-titres dédié doit être développé.

3.4.7 Changement d'espace colorimétrique

La DCSS définit comme espace colorimétrique le CIE 1931 XYZ. Les espaces colorimétriques utilisés pour le rendu sur écran ou la projection sont des sous-ensembles du CIE 1931 XYZ. La DCSS laisse donc à l'équipement de rendu des images le soin de faire la conversion. Pour les projecteurs conformes aux exigences du DCI, la valeur imposée pour le gamma est 2,6.

Nous avons adapté VLC à l'espace colorimétrique XYZ, et écrit une fonction de conversion du format XYZ au format sRGB pour l'affichage vidéo. Pour que le rendu des couleurs soit correct sur du matériel non conforme au DCI, nous effectuons une conversion de l'espace colorimétrique XYZ vers l'espace sRGB avec un gamma égal à 2,2, afin d'obtenir des palettes de couleurs plus standard pour les écrans d'ordinateurs et certains projecteurs.

La conversion en sRGB doit être appliquée à chaque pixel de l'image après son décodage. Afin de permettre une exécution rapide de cette fonction, nous l'avons incluse dans le module OpenGL de VLC, en écrivant cette fonction en *OpenGL Shading Language* (GLSL) pour qu'elle soit exécutée sur la carte graphique. Ceci nous permet d'accélérer les traitements en utilisant les co-processeurs graphiques.

3.5 Évaluation

L'évaluation de la lecture d'un DCP porte sur l'intégration du module dans le logiciel VLC et la méthode mise en place pour le faire accepter. Ensuite, nous évaluons la lecture d'un DCP.

3.5.1 Validation du module DCP dans VLC

Nous avons écrit un module de lecture de DCP dans VLC. Ce code n'est pas resté sur nos machines, mais a été publié dans VLC¹. Pour publier le code dans la branche officielle de VLC, il faut respecter les règles de développement de la communauté VLC.

Pour pouvoir publier un module dans VLC, il faut proposer un ou plusieurs *patches* de code. Les *patches* sont relus et commentés par les membres de la communauté. Suivant la taille du *patch*, et notamment pour la création de nouveaux modules, le processus de publication peut être long. Avant de soumettre un *patch* de module entier, on soumet d'abord un pré-*patch* appelé *Request For Comment* (RFC). À travers le RFC, on demande un avis à la communauté sur le *patch* que l'on soumettra. Les relecteurs évaluent la qualité du codage au niveau technique : implémentation correcte des fonctionnalités du module au regard de l'architecture de VLC. Le module relu est également évalué au niveau de la qualité du code : respect de règles de codage, clarté du code écrit et aptitude à être mis à jour (*maintainability*).

Le premier RFC pour le module DCP a été soumis le 3 juin 2013, et l'intégration officielle du module dans VLC a eu lieu le 15 décembre 2013. Il a donc fallu plus de six mois pour faire accepter le module. Dans cet intervalle, nous avons soumis deux RFC, et plus de seize versions du *patch* du module DCP (cf. figure 3.5). En effet, après chaque soumission de *patch*, les relecteurs faisaient des commentaires, la prise en compte des commentaires était discutée, puis un nouveau *patch* était soumis : il a fallu ces seize itérations pour faire accepter le module. Notons que le premier *patch* soumis ne comprenait que la lecture d'un DCP avec une seule bobine virtuelle pour la vidéo et

1. <http://git.videolan.org/?p=vlc.git;a=tree;f=modules/access/dcp>

l'audio : il ne comprenait ni la lecture de plusieurs bobines, ni la gestion des sous-titres, ni le chiffrement. Ces fonctions ont été ajoutées progressivement, en soumettant de nouveaux *patches* sur le module DCP.

Nous nous sommes confrontés à une difficulté inattendue dans le développement du module : la compatibilité entre licences libres. VLC a une licence *GNU General Public License* (GPL) en version 2, donc notre module aussi. Nous utilisons la bibliothèque ASDCPlib pour les fichiers MXF. Cette bibliothèque a une licence de type « New BSD license », compatible avec la licence GPL en version 2. En revanche, ASDCPlib utilise la bibliothèque OpenSSL, qui a une licence « Apache License 1.0 ». Or, cette licence Apache n'est pas compatible avec la licence GPL en version 2 de VLC, en raison d'une clause dite « de publicité ». Chaque logiciel utilisant OpenSSL doit ajouter la phrase « *This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit* » lors de la promotion ou de la redistribution. Cette restriction rend la licence d'OpenSSL incompatible avec la licence GPL en version 2 de VLC. La clause de publicité est incompatible, car la licence GPL n'autorise pas l'ajout de restrictions en plus de celles déjà imposées. En pratique, l'utilisation de licence avec clause de publicité est difficile à mettre en place. Dans le cas de VLC, qui utilise plusieurs centaines de bibliothèques, il faudrait ajouter lors de chaque publicité ou distribution de VLC, une phrase pour chaque bibliothèque utilisée².

ASDCPlib n'utilise OpenSSL que pour la partie chiffrement AES. La solution à ce problème de licence consisterait à recompiler ASDCPlib avec une bibliothèque gérant le chiffrement AES, qui soit compatible avec la licence GPL, comme par exemple la bibliothèque gcrypt.

La figure 3.5 comporte des références à des scripts LUA (langage de script utilisé par VLC). Avant de commencer le développement du module, nous avons testé la lecture de DCP par ce langage. Cela aurait permis de lire des DCP sans avoir à créer un module spécifique. Cette solution s'est vite avérée trop contraignante, car elle ne nous permettait pas de gérer le chiffrement des DCP, ni la lecture de plusieurs bobines virtuelles.

Nous pouvons également voir, sur la figure 3.5, les contributions postérieures à l'acceptation du PATCHv16 : il y a les contributions pour la lecture de plusieurs bobines (*Read multiple reels*), ainsi que celles pour le chiffrement (*read encrypted DCPs*). Certains *patches* correspondent à des corrections de *bugs*.

On peut noter que le *patch* pour la lecture des sous-titres (*allow to read subtitles*) est dans l'état *Changes Requested*. La lecture de sous-titres n'est pas complètement intégrée dans la branche officielle. Le *patch* est disponible dans notre propre branche de VLC³, mais le code n'est pas encore assez mature pour être accepté dans VLC. Nous devons fournir un code plus robuste et plus apte à être mis à jour (*maintainable*). Le module de gestion des sous-titres est proposé dans une version simplifiée, c'est-à-dire qu'il ne s'agit pas d'un décodeur dédié, mais de code inséré dans le module DCP. Nous n'avons pas implémenté la gestion des sous-titres sous forme d'images PNG, car cette possibilité, qui est très peu utilisée, est supplantée par l'utilisation de sous-titres au format XML.

2. <https://www.gnu.org/philosophy/bsd.html>

3. Dépôt git du projet : <git://utopialab.tetaneutral.net/home/git/projects/vlc.git>

Patchwork VLC		login	register	mail	settings	about
Project: vlc-devel · patches · project info · other projects						
Incoming patches						
Patch	Date	Submitter	Delegate	State		
dcp: Creation of LUA script to read DCP	2013-03-05	Pierre Villard		Superseded		
LUA script to read DCP	2013-03-06	Pierre Villard		Superseded		
LUA script to read DCP	2013-03-08	Pierre Villard		Superseded		
[RECI] dcp: Creation of access-demux module for DCP (using asdcpilib)	2013-06-03	Simona-Marinela Prodea		Rejected		
[RECI] dcp: Creation of access-demux module for DCP (using asdcpilib)	2013-06-25	Simona-Marinela Prodea		Superseded		
dcp: Creation of access-demux module for DCP (using asdcpilib)	2013-07-12	Nicolas BERTRAND		Superseded		
IV21 dcp: Creation of access-demux module for DCP (using asdcpilib)	2013-07-18	Nicolas BERTRAND		Rejected		
IV31 dcp: Creation of access-demux module for DCP (using asdcpilib)	2013-09-02	Nicolas BERTRAND		Changes Requested		
IV51 dcp: Creation of access-demux module for DCP	2013-10-07	Nicolas BERTRAND		Changes Requested		
[PATCHv6] dcp: Creation of access-demux module for DCP	2013-11-22	Nicolas BERTRAND		Superseded		
IV71 dcp: Creation of access-demux module for DCP	2013-11-27	Nicolas BERTRAND		Superseded		
[PATCHv9] dcp: Creation of access-demux module for DCP	2013-11-29	Nicolas BERTRAND		Superseded		
[PATCHv10] dcp: Creation of access-demux module for DCP	2013-12-02	Nicolas BERTRAND		Superseded		
[PATCHv11] dcp: Creation of access-demux module for DCP	2013-12-03	Nicolas BERTRAND		Superseded		
[RECI] dcp: Audio channels reordering	2013-12-08	Nicolas BERTRAND		Superseded		
[RECV2] dcp: Audio channels reordering	2013-12-09	Nicolas BERTRAND		Superseded		
[PATCHv12] dcp: Creation of access-demux module for DCP	2013-12-10	Nicolas BERTRAND		Superseded		
[PATCHv13] dcp: Creation of access-demux module for DCP	2013-12-10	Nicolas BERTRAND		Superseded		
[PATCHv14] dcp: Creation of access-demux module for DCP	2013-12-11	Nicolas BERTRAND		Superseded		
[PATCHv15] dcp: Creation of access-demux module for DCP	2013-12-11	Nicolas BERTRAND		Superseded		
[PATCHv16] dcp: Creation of access-demux module for DCP	2013-12-13	Nicolas BERTRAND		Accepted		
dcp: Integrate asdcpilib with .pc files	2014-02-04	Nicolas BERTRAND		Superseded		
dcp: Integrate asdcpilib with .pc files	2014-02-04	Nicolas BERTRAND		Accepted		
dcp: Command-line option to choose a CPL	2014-02-06	Valentin Vetter		Changes Requested		
[PATCH/RECI] dcp: Read multiple reels	2014-02-19	Valentin Vetter		Accepted		
DCP: pause	2014-02-26	Valentin Vetter		Accepted		
DCP: allow to read subtitles	2014-03-01	Valentin Vetter		Changes Requested		
dcp: read encrypted DCPs	2014-03-31	Simona-Marinela Prodea		Changes Requested		
[PATCHv2] dcp: read encrypted DCPs	2014-04-18	Simona-Marinela Prodea		Changes Requested		
dcp: read encrypted DCPs	2014-04-29	Simona-Marinela Prodea		Superseded		
IV41 dcp: read encrypted DCPs	2014-04-29	Simona-Marinela Prodea		Accepted		
dcp: ignore prefixes in XML nodes	2014-05-14	Simona-Marinela Prodea		Accepted		
dcp: Add case GET PTS_DELAY in Control()	2014-05-23	Nicolas BERTRAND		Accepted		

FIGURE 3.5 – Ensemble de *patches* relatifs au module DCP dans VLC (extrait de https://patches.videolan.org/project/vlc-devel/list/?q=dcp&state=*%&archive=both&order=date).

Intitulé du <i>patch</i> dans VLC
OpenGL: Addition of XYZ to sRGB conversion in GLSL
OpenGL: take planes in account in DrawWithShaders
avcodec: Add XYZ colorspace
add XYZ colorspace/fourcc

TABLE 3.2 – *Patches* relatifs à la gestion par VLC de l’espace colorimétrique XYZ.

Nous avons également contribué à ajouter dans VLC l’espace colorimétrique XYZ et sa conversion en sRGB. L’ensemble des *patches* est présenté dans la table 3.2. Les modifications sur l’espace colorimétrique XYZ ont consisté à déclarer ce nouvel espace dans VLC. Dans le module OpenGL, nous avons écrit les matrices permettant la conversion en sRGB et le changement de gamma. À l’avenir, nous pouvons imaginer un changement d’espace colorimétrique et de gamma paramétrable. Cela pourrait être utile pour adapter le module DCP à un plus grand nombre de dispositifs de projection ou d’écrans.

3.5.2 Évaluation de la lecture d’un DCP

Évaluer la performance de lecture d’un DCP consiste à vérifier que le décodage des images, celui du son et celui des sous-titres sont bien synchronisés tout au long de la lecture. VLC fournit un ensemble de messages d’alerte pour indiquer des problèmes de lecture dans le flux. Notamment, il peut indiquer qu’un décodeur est trop lent (i.e., qu’il ne tient pas la cadence de 24 images par seconde) ou bien qu’il y a des pertes de synchronisation entre les flux. Nous avons également utilisé l’inspection visuelle pour évaluer la lecture de DCP : nous vérifions à chaque lecture de DCP que l’image, le son et les sous-titres sont synchronisés et que le contenu des flux est correctement restitué.

Les films-annonces utilisés pour l’évaluation sont disponibles ici : <http://utopialab.tetaneutral.net/DCPsamples>.

Film-annonce	Débit binaire (Mb/s)	Résolution 2K	2K interpolé depuis 1K
<i>Spring Breakers</i>	201	Non	Oui
<i>Django Unchained</i>	151	Non	Oui
<i>Moonrise Kingdom</i>	128	Non	Oui
<i>Jeux d’été</i>	79	Non	Oui

TABLE 3.3 – Lecture de DCP sur portable i7 (8 cœurs).

Les résultats de l’évaluation de la lecture sont reportés dans la table 3.3 pour la lecture sur un ordinateur portable i7, et dans la table 3.4 pour la lecture sur une station de travail Bi-Xeon. On mesure ici la performance globale de la lecture, une évaluation plus détaillée du décodeur JPEG2000 étant faite dans le chapitre 4.

Film-annonce	Débit binaire (Mb/s)	Résolution 2K	2K interpolé depuis 1K
<i>Spring Breakers</i>	201	Non	Oui
<i>Django Unchained</i>	151	Non	Oui
<i>Moonrise Kingdom</i>	128	Oui	Oui
<i>Jeux d'été</i>	79	Oui	Oui

TABLE 3.4 – Lecture de DCP sur station de travail Bi-Xeon (24 cœurs).

Dans les résultats présentés ici, nous n'arrivons pas à lire un DCP en 2K, même si nous arrivons à décoder les images JPEG2000 à un débit supérieur à 24 images par seconde (cf. chapitre 4). Il existe plusieurs raisons à cette limitation. Tout d'abord, VLC limite à 16 *threads* le décodeur libavcodec, mais même si l'on désactive cette limitation, on rencontre le même problème. L'autre limitation est due à la profondeur de bits des couleurs, qui sont codées sur 12 bits. On utilise donc trois mots de 16 bits pour coder un pixel. Ce sont la transmission des données vers le module OpenGL et les traitements non optimisés pour des données de 16 bits qui limitent l'affichage des images en 2K.

Nous pouvons en revanche lire correctement un DCP décompressé à une résolution de 1K, puis l'interpoler en 2K. L'exécution de l'interpolation n'a pas d'impact sur la lecture du DCP : l'affichage des images ne reste pas bloqué, et il n'y a pas de désynchronisation entre les flux.

Pour la lecture de certains DCP, nous avons rencontré des problèmes de « gel » de l'affichage vidéo. Les informations enregistrées par VLC indiquaient le problème suivant :

```
1 decoder/packetizer fifo full (data not consumed quickly enough), resetting fifo!
```

Cette alarme est déclenchée par le noyau de VLC. Elle indique que la FIFO entre le *demuxer* et le décodeur est pleine. Le décodeur ne consomme pas assez vite les données envoyées par le module DCP pour les images. Le noyau prend la décision de vider cette FIFO : par conséquent, l'affichage de certaines séquences d'images est perdu. La taille de la FIFO est limitée à 400 Mo, ce qui correspond à 30 secondes de contenu avec un débit binaire de 100 Mb/s. Le noyau de VLC impose cette contrainte pour ne pas avoir un usage de RAM trop élevé, et éviter ainsi de bloquer le système d'exploitation lors d'un problème de lecture de vidéo.

Un DCP peut avoir un débit binaire supérieur à 100 Mb/s (jusqu'à 250 Mb/s). Pour éviter que la FIFO ne se remplisse trop vite avec un DCP ayant un débit binaire élevé, il faut réduire la durée du cache de données lors de la lecture sur le disque dur. Avec l'option de VLC `--file-caching`, on peut réduire la taille de ce cache. En la réduisant à 10 secondes, nous n'avons plus rencontré de problème de FIFO pleine.

3.6 Bilan

Nous avons expliqué dans ce chapitre les raisons ayant conduit au choix du lecteur multimédia VLC pour la lecture de DCP. VLC a été choisi car il s'agit d'un lecteur multimédia libre, très utilisé et supportant un grand nombre de codecs vidéo et audio. Cela nous a donc évité de développer un lecteur de DCP dédié.

Un module de gestion de DCP a été créé dans le code de VLC. Il permet de lire des DCP avec images, son et sous-titres. Il gère la lecture de DCP contenant plusieurs bobines virtuelles. On peut aussi lire des DCP chiffrés, le détail de l'implémentation étant présenté dans le chapitre 5.

Le module de gestion de DCP est officiellement disponible dans VLC. Il n'est pas distribué dans toutes les versions de VLC, car nous rencontrons des problèmes de compatibilité de licences dus à l'utilisation de la bibliothèque OpenSSL dans libavcodec.

Le traitement de l'espace colorimétrique CIE XYZ a également été pris en compte dans le lecteur, que nous convertissons par défaut au format sRGB avec un gamma égal à 2,2.

La lecture de DCP ne peut pas encore être faite en pleine résolution (2K), car nous rencontrons des limitations de performance au niveau du décodage et du traitement des images 2K sur 12 bits. Néanmoins, nous arrivons à lire des DCP en les décodant jusqu'au format 1K, puis en réalisant une interpolation en 2K.

Dans le prochain chapitre, nous décrivons l'implémentation du décodeur JPEG2000 développé dans libavcodec.

Chapitre 4

Un décodeur JPEG2000 dédié au cinéma numérique

4.1 Introduction

Pour jouer un DCP, il nous faut atteindre un débit de décodage d'au moins 24 images par seconde, et synchroniser les images, le son et les sous-titres. La principale difficulté pour atteindre ce but est la complexité du codage JPEG2000.

Le JPEG2000 a été choisi par le DCI comme format pour la compression des images. Le cinéma numérique impose l'utilisation d'images de grande taille et de très bonne qualité, ce qui implique de stocker et de manipuler une grande quantité de données. Il est donc nécessaire de compresser les images, sans quoi le DCP devant être diffusé vers les salles serait trop volumineux (de l'ordre de plusieurs To). La compression des images doit être efficace sans provoquer trop de pertes. Le format JPEG2000, qui est adapté aux images de grande taille et de précision élevée, respecte ces contraintes. Il est également nécessaire, pour le montage de films, que l'on puisse facilement manipuler les images. Pour cela, le DCI impose un codage image par image, c'est-à-dire sans compression inter-frames. Le JPEG2000, qui est un codeur d'images, vérifie tout naturellement cette autre contrainte. Le JPEG2000 a la propriété d'être robuste aux changements d'échelle (*scalability*) : grâce à l'utilisation de la transformation en ondelettes, l'image peut être décompressée à plusieurs résolutions. Cette propriété est très pratique pour le montage de films : cela permet d'effectuer le montage d'un film avec les images de faible définition, puis de générer le *master* en haute définition. Le format JPEG2000 a été choisi par le DCI pour toutes ces raisons.

Dans ce chapitre, nous présentons le fonctionnement du format JPEG2000. La norme JPEG2000 [20] spécifie des profils pour le cinéma numérique, dont nous explicitons le détail. Nous montrons ce que ces profils impliquent pour l'implémentation d'un décodeur. Nous présentons enfin les différents types de décodeurs JPEG2000 existants et les raisons qui nous ont amenés à développer notre propre décodeur.

4.2 Le format JPEG2000

Le JPEG2000 correspond à la partie 1 du standard ISO 15444, qui a été publié en 2000. Ce format comporte un mode sans perte et un autre avec pertes.

Bien que nous ayons uniquement à nous préoccuper du décodage d'images JPEG2000, nous présentons ici le format JPEG2000 du point de vue du codage. Ce codage permet de passer d'une image à un fichier binaire compressé. Le taux de compression est un paramètre clé du codage, mais au moment du décodage, lorsque nous devons décoder un fichier binaire compressé au format JPEG2000, ce taux de compression nous sera imposé.

Nous présentons ci-après les étapes du codage par l'algorithme JPEG2000. Le décodage d'une image JPEG2000 consistera à suivre les mêmes étapes dans l'ordre inverse et à appliquer, à chaque étape, la transformation inverse. Mais si certaines étapes du codage ne sont pas inversibles (par exemple la quantification), cela causera inévitablement des pertes au moment du décodage.

4.2.1 Découpage en tuiles et changement d'espace colorimétrique

L'élément de départ à coder est l'image, qui est divisée en plusieurs parties appelées « tuiles » (*tiles*). Une tuile correspond à une zone spatiale de l'image. Une image peut être constituée d'une ou de plusieurs tuiles.

Les images couleur comportant trois composantes, il en va de même pour les tuiles. Avant d'appliquer une transformation en ondelettes à chaque composante couleur d'une tuile, appelée « composante de tuile » (*tile-part*), on procède à un changement d'espace colorimétrique. En l'occurrence, on passe de l'espace RGB (rouge-vert-bleu) à l'espace YCbCr (luminance-chrominances), et ce afin de décorréler les canaux. En effet, il est connu que, de manière générale, la compression de données est d'autant meilleure que les données sont moins corrélées. Il existe deux types de changements d'espace colorimétrique, inversible (*Reversible Component Transformation*) ou non inversible (*Irreversible Component Transformation*), qui sont utilisées, respectivement, pour le codage sans perte ou avec pertes.

4.2.2 Transformation en ondelettes discrète

La transformation en ondelettes discrète, ou *Discrete Wavelet Transform* (DWT), existe elle aussi sous deux formes : inversible ou non inversible. Les ondelettes 5/3 de Le Gall sont utilisées pour la compression sans perte. Pour atteindre de meilleures performances de compression (avec pertes), les ondelettes 9/7 de Daubechies sont utilisées [5]. Dans les deux cas, la partie 1 de la norme JPEG2000 impose l'utilisation de filtres de type *lifting-based*. L'algorithme précis pour la transformation en ondelettes discrète (inversible ou non inversible) est décrit dans le chapitre F2 de [20]. Enfin, toutes les DWT dont il est question sont bidimensionnelles.

La figure 4.1 illustre les premières étapes du codage JPEG2000 : sur cet exemple, deux DWT successives sont appliquées à la première composante (canal Y) d'une tuile

de l'image d'origine. Par souci de clarté, nous avons agrandi le résultat des DWT, alors que sa taille est exactement la même que celle de la composante de tuile dont il provient. En d'autres termes, l'ensemble des « coefficients d'ondelettes » obtenus en appliquant deux DWT à l'ensemble des composantes des différentes tuiles (ici, il y a quatre tuiles) a exactement la même taille que l'image d'origine.

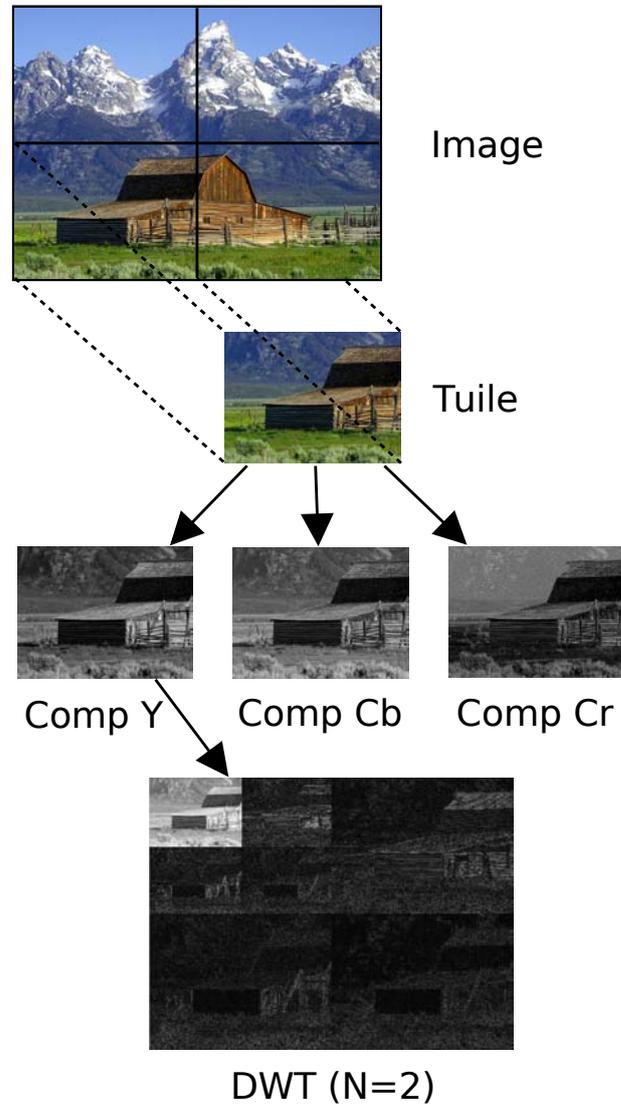


FIGURE 4.1 – Premières étapes du codage JPEG2000.

La DWT est utilisée pour obtenir plusieurs « niveaux de décomposition » : une DWT donne une image de définition quatre fois plus faible. Chaque niveau de décomposition comporte trois « sous-bandes » fréquentielles, notées x_{HL} , x_{LH} et x_{HH} au niveau de

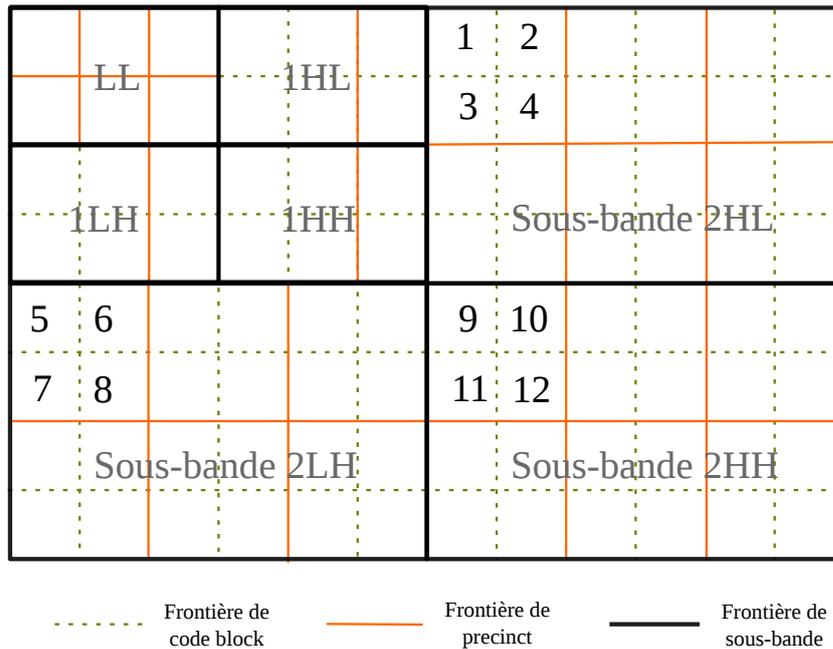


FIGURE 4.2 – *Code blocks* et *precincts* après application de deux DWT à une composante de tuile. Cet exemple comporte trois niveaux de décomposition. La taille d'un *code block* est toujours la même, mais celle d'un *precinct* peut dépendre du niveau de décomposition. Par exemple, les six *precincts* de la sous-bande LL sont plus petits que ceux des niveaux supérieurs : il sont composés chacun d'un seul *code block*.

décomposition x , à l'exception du niveau de décomposition 0 (basses fréquences), qui comporte une seule sous-bande notée LL. La figure 4.2 donne l'exemple d'une composante de tuile à laquelle deux DWT ont été appliquées. Chaque niveau de décomposition contient des sous-bandes composées de coefficients qui décrivent les caractéristiques en fréquences spatiales (verticales et horizontales) de la composante de tuile.

Après application des transformations en ondelettes, chaque sous-bande est découpée en *code blocks*. Il s'agit d'éléments rectangulaires, dont chacune des dimensions est égale à une puissance de 2. Un *precinct* est un ensemble de *code blocks* correspondant aux différentes sous-bandes de même niveau de décomposition, couvrant une même zone spatiale. Sur l'exemple de la figure 4.2, les *code blocks* 1, 2, 3, 4 de la sous-bande 2HL sont regroupés avec les *code blocks* de numéros 5 à 8 de la sous-bande 2LH et avec les *code blocks* de numéros 9 à 12 de la sous-bande 2HH, pour former un *precinct*. On voit également qu'un *precinct* de seulement six *code blocks* est construit à partir des sous-bandes 1HL, 1LH et 1HH, à cause de la forme rectangulaire de la tuile.

4.2.3 Quantification des coefficients d'ondelettes

La quantification est un procédé qui, de façon générale, diminue la précision des données. Elle permet de réduire la dynamique des coefficients d'ondelettes, et donc d'améliorer la compression. Pour le codage sans perte, il ne doit bien sûr pas y avoir de quantification.

Pour le codage avec pertes, les coefficients sont quantifiés par la méthode *dead zone scalar quantifier*, qui consiste à mettre à 0 les coefficients inférieurs en valeur absolue à un seuil, afin de réduire la dynamique des données (voir [38], page 43).

4.2.4 Codage entropique des *code blocks* (*Tier-1*)

Lors des transformations en ondelettes, les pixels ne sont pas compressés : ils sont réorganisés de manière fréquentielle afin d'atteindre de meilleurs taux de compression. C'est ce qui est fait à l'étape *Tier-1*¹ : à chaque *code block* est appliqué un « codage entropique », indépendamment des autres *code blocks*.

L'étape *Tier-1* utilise un codeur arithmétique (codeur MQ) et un codeur contextuel « par plan de bits » (EBCOT). L'algorithme *Embedded Block Coding with Optimized Truncation* (EBCOT) a été proposé par Taubman en 2000 [48]. Son fonctionnement étant complexe, nous renvoyons le lecteur à [49] pour une description détaillée.

4.2.5 Création du fichier binaire (*Tier-2*)

Cette étape permet de contrôler le taux de compression et d'organiser le « train de bits » (*bitstream*) du format JPEG2000. Au moment de la compression, le train de bits peut être tronqué de différentes façons, ce qui permet de choisir le taux de compression.

Le fichier binaire est organisé de manière hiérarchique, de l'image vers les *code blocks*. À chaque niveau, il est composé de deux parties : un en-tête et des données. Au plus haut niveau, il comporte l'en-tête principal et les données correspondant aux différentes composantes de tuiles, qui elles-mêmes comportent chacune un en-tête, suivi de données compressées.

L'en-tête principal du fichier binaire et celui d'une composante de tuile sont construits selon le même modèle : ils contiennent des marqueurs indiquant les paramètres de l'image (taille, nombre de composantes, ...). Ces marqueurs peuvent être obligatoires ou optionnels.

Les données compressées de chaque composante de tuile sont organisées en « paquets ». Un paquet correspond aux données associées à un *precinct*. L'élément le plus petit, dans la création du fichier binaire, est le *code block*. Le train de bits associé à chaque *code block* contient les données binaires issues de la compression du codeur entropique.

La figure 4.3 met en correspondance la décomposition de l'image effectuée par le codage JPEG2000 et l'organisation des données compressées dans le fichier binaire associé.

1. Le mot *tier* signifie « niveau » en anglais.

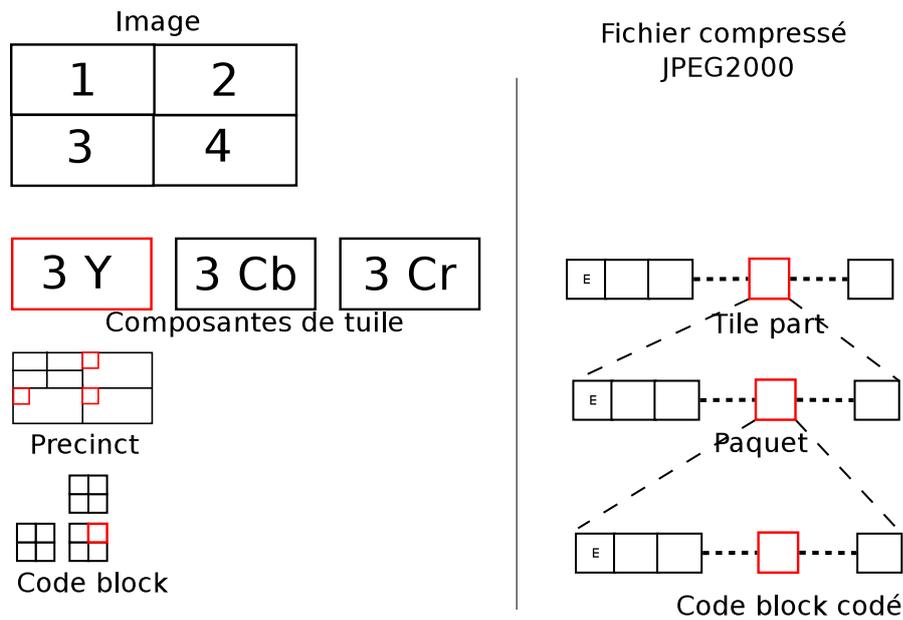


FIGURE 4.3 – Correspondances entre la décomposition de l’image effectuée par le codage JPEG2000 et l’organisation des données compressées dans le fichier binaire (E signifie « en-tête »).

4.2.6 Décompression du format JPEG2000

Lors du décodage d’une image compressée au format JPEG2000, le train de bits correspondant à chaque *code block* est extrait du fichier binaire par le *Tier-2*, qui extrait également du train de bits le nombre de « passes de codage » de ce *code block*. Le *Tier-1*, qui reçoit les données compressées fournies par le *Tier-2*, effectue la décompression à proprement parler (cf. figure 4.4).

Les coefficients d’ondelettes de chaque *code block* sont décompressés par « décodage entropique ». Le principe de fonctionnement du *Tier-1* est le même pour le codage et pour le décodage : EBCOT code indépendamment chaque *code block* d’un paquet. Le décodeur entropique arithmétique adaptatif binaire basé sur le contexte (*context-based adaptive binary arithmetic*), appelé « décodeur MQ », est issu du standard ISO JBIG2.

4.3 Contraintes du cinéma numérique sur le JPEG2000

Des options pour l’encodage JPEG2000 sont imposées depuis 2004 par le DCI, qui a choisi JPEG2000 comme format de compression. Le DCI impose que la compression soit effectuée image par image. Il impose également qu’il y ait une seule tuile par image,

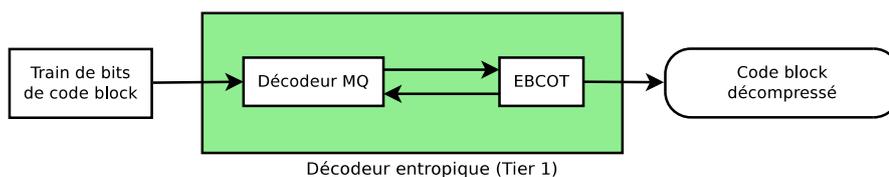


FIGURE 4.4 – Décodage entropique du JPEG2000.

afin d'éviter les artéfacts aux frontières des tuiles². Le DCI a souhaité avoir un format ouvert [7], qui supporte l'encodage des images avec une profondeur de bits importante (12 bits) et qui supporte le format de l'espace colorimétrique CIE 1931 XYZ utilisé par le cinéma numérique. Les auteurs de [7] citent les contraintes imposées par le DCI sur le JPEG2000, sans expliciter les raisons de ce choix : c'est ce que nous nous proposons de faire dans ce paragraphe.

4.3.1 Contraintes sur la transformation en ondelettes

Le DCI a choisi d'utiliser la compression avec pertes. Avec la compression sans perte, le fichier correspondant à l'ensemble des images d'un film aurait été très volumineux, ce qui aurait compliqué sa diffusion. Dans le codage avec pertes, on utilise une DWT irréversible. Le DCI réglemente cette perte en imposant que, lors du décodage, la DWT inverse utilise au moins 16 bits de précision, car le calcul est fait en flottants.

Un fichier image doit pouvoir comporter les définitions 2K (2048×1080) ou 4K (4096×2160). Le DCI n'impose pas l'utilisation de la DWT pour le 2K, mais impose toutefois au moins une DWT pour le format 4K, afin de pouvoir en extraire le format 2K. Cette contrainte permet de garantir qu'il est toujours possible de projeter une copie numérique au format 4K sur un projecteur 2K.

Le DCI fixe aussi le nombre maximal de DWT à cinq pour le 2K³, et à six pour le 4K. Pour un même taux de compression, l'utilisation de plusieurs DWT permet de décorréler les données, donc de limiter les pertes (la qualité d'image sera meilleure à la projection). Bien entendu, cela augmente aussi le temps de calcul lors du décodage.

Le DCI spécifie que le décodeur ne doit ignorer aucun niveau de décomposition pour gérer des pics de charge lors du décodage. Nous pourrions nous permettre un écart dans le cas du projecteur alternatif, si nous montrons que la dégradation de l'image n'est pas perceptible à la projection (cf. paragraphe 4.6).

2. La décomposition en tuiles est surtout utile pour les très grandes images, comme par exemple les images satellitaires.

3. En pratique, tous les DCP que nous avons manipulés correspondaient à ce cas de figure : format 2K comportant six niveaux de décomposition (après application de cinq DWT).

4.3.2 Contraintes sur la compression

Le DCI impose que chaque *code block* soit de taille 32×32 . La taille d'un *precinct* doit être 256×256 , sauf pour la bande de fréquences la plus faible (LL), où cette taille est 128×128 , et éventuellement aux frontières des sous-bandes (cf. figure 4.2).

La taille maximale d'une image 2K compressée, en-têtes compris, est égale à 1302083 octets. À titre de comparaison, à la fréquence d'affichage (*frame rate*) standard de 24 images par seconde, la valeur 1302083 correspond à un « débit binaire » (*bit rate*) maximal égal à 250 Mb/s, ce qui représente une valeur élevée par rapport aux débits binaires du MPEG2 HD (50 Mb/s) et du MPEG2 DVD (9 Mb/s). Avec un tel débit, le DCI a choisi d'avoir la meilleure qualité d'image possible, au détriment de l'espace disque. Effectivement, la copie numérique d'un film peut occuper de 200 à 300 Go pour le format 2K, et jusqu'à plus de 500 Go pour le format 4K.

4.3.3 Contraintes sur la formation du train de bits

Le format JPEG2000 autorise plusieurs ordres de parcours pour la décomposition de l'image, mais le DCI impose un ordre bien spécifique. Cet ordre correspond à l'ordre de parcours *component-position-resolution-layer* autorisé par le format JPEG2000, si ce n'est que le dernier étage est escamoté, car le DCI impose qu'il y ait une seule « couche de qualité » (*layer*). Cela signifie que, lors du codage, les *precincts* sont organisés d'abord par composante couleur, puis par position du *precinct* dans toutes les sous-bandes. Les sous-bandes sont parcourues dans l'ordre LL, 1HL, 1LH, 1HH, 2HL, . . . NHH (cf. figure 4.5). Le DCI impose donc qu'une image soit décompressée selon un parcours spatial et qu'elle soit affichée totalement en une seule fois, afin de ne pas avoir d'effet de transition entre deux images du flux vidéo.

Quatre marqueurs (*coding style default*, *coding style component*, *quantization default* et *quantization component*) ne peuvent apparaître que dans l'en-tête principal du fichier : tous les *code blocks* d'une image sont donc traités de la même façon, vis-à-vis de ces marqueurs (voir [20], page 41).

4.4 Solutions de décodage du JPEG2000

Il n'existe à ce jour aucun logiciel libre permettant de décompresser en temps réel des images JPEG2000 au format du cinéma numérique. Le logiciel Kakadu (version 6.0) réalisé par Taubman et ses collaborateurs est censé décoder les images au format 2K avec un rapport d'aspect de 2,39 (2048×857) et un débit binaire de 244 Mb/s, à une fréquence de 24 images par seconde, sur des ordinateurs à huit cœurs. Notons que ce rapport d'aspect correspond aux images les plus petites du format 2K. Notre objectif est de réaliser un logiciel de décompression libre, ce qui n'est pas le cas de Kakadu.

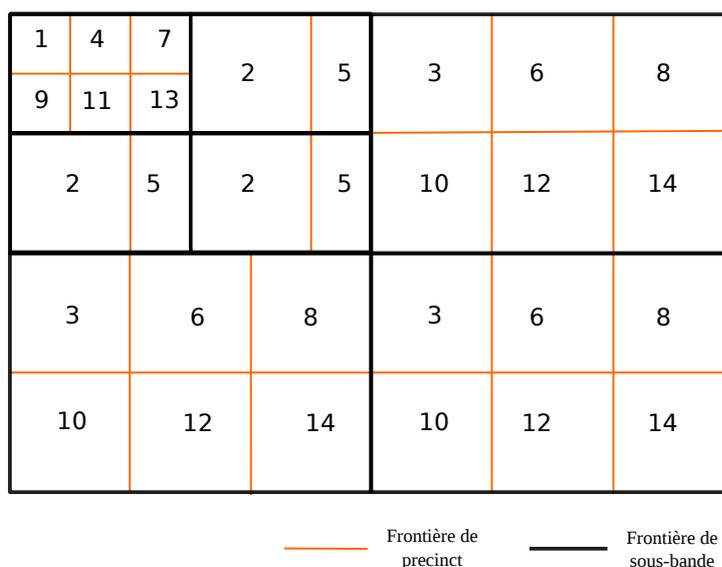


FIGURE 4.5 – Les numéros indiquent dans quel ordre les *precincts* sont organisés dans le fichier binaire, lors du codage.

4.4.1 Décodage matériel du JPEG2000

Des architectures matérielles spécifiques fondées sur des circuits de type *Field-Programmable Gate Array* (FPGA) ont été proposées. Dans [17], les auteurs présentent une méthode de décompression JPEG2000 par FPGA pour le cinéma numérique. Dans [37], une architecture conforme à la spécification du DCI est présentée, qui non seulement permet de décoder le format JPEG2000, mais effectue le déchiffrement du DCP et le marquage du contenu à projeter (cf. chapitre 5).

Il existe également des puces spécifiquement dédiées à la compression/décompression du JPEG2000, comme la puce ADV202 de la société Analog Devices. Une architecture matérielle utilisant cette puce est présentée dans [50]. Dans les serveurs actuellement installés dans les cinémas, on retrouve ces cartes électroniques dédiées à la décompression JPEG2000. Il s'agit de la solution la plus utilisée.

Nous n'utilisons pas ces solutions matérielles qui, d'une part, sont trop chères, et d'autre part, nécessitent l'utilisation de matériel spécifique, ce qui ne correspond pas à notre souhait de développer un outil le plus générique possible.

4.4.2 Décodage logiciel du JPEG2000

L'algorithme de codage du JPEG2000 est complexe. Le décodage, qui est essentiellement la fonction inverse du codage, est donc complexe lui aussi, en particulier si l'on vise le temps réel (décompression des images à une fréquence au moins égale à 24 images par seconde) pour les images de haute définition du cinéma numérique.

Pour pouvoir atteindre cette fréquence de décodage, il est nécessaire de recourir à des méthodes de calcul en parallèle. Il existe plusieurs solutions pour le décodage logiciel du JPEG2000, qui utilisent l’une des trois approches suivantes :

- L’utilisation de *threads* au niveau CPU, combinée avec des jeux d’instructions spécifiques au processeur, telles que les *Streaming SIMD Extensions* (SSE) .
- L’utilisation des processeurs des cartes graphiques par calculs sur *General-purpose Processing on Graphics Processing Units* (GPGPU).
- Une combinaison de ces deux approches.

Dans le JPEG2000, le décodage entropique (EBCOT) est la partie la plus complexe, et aussi celle qui prend le plus de temps de calcul. La parallélisation du décodeur entropique est délicate, car la structure d’entrée (le train de bits) est découpée en blocs de données difficiles à traiter en parallèle. Le *code block* est la plus petite structure que l’on puisse segmenter dans le train de bits, et chaque *code block* est codé indépendamment des autres. On peut donc envisager un traitement en parallèle de tous les *code blocks* d’une image, mais le nombre d’opérations à faire pour chaque *code block* étant élevé, cela rend le traitement en parallèle des *code blocks* peu efficace à cause de problèmes de « latence ». En réalité, dans tous les articles où un traitement en parallèle est envisagé pour le décodage du JPEG2000, comme dans [26] et [31], où des méthodes pour paralléliser l’algorithme EBCOT en GPGPU sont décrites, la parallélisation concerne les traitements au niveau d’un seul *code block*.

Certaines solutions logicielles existantes sont « propriétaires », d’autres sont libres. Les principaux outils pour le décodage JPEG2000 sont regroupés dans la table 4.1.

Logiciel	Type de licence	Type de processeur
EasyDCP [12]	Propriétaire	GPGPU
Kakadu [47]	Propriétaire	CPU
JasPer [3]	Libre	CPU
OpenJPEG [1]	Libre	CPU

TABLE 4.1 – Principaux outils de décodage logiciel du format JPEG2000.

Le logiciel EasyDCP est un lecteur de DCP breveté [12] réalisant le décodage en GPGPU. Une approche *Group Of Pictures* (GOP), qui utilise la similarité des *code blocks*, permet d’éviter la latence entre processeurs du GPGPU.

Taubman [47] montre qu’il est possible d’atteindre un décodage à un débit de 24 images par seconde en utilisant le *multithreading*, ainsi que des instructions SSE dont le but est d’optimiser le code. Cette solution est implémentée dans le logiciel « propriétaire » Kakadu.

Les auteurs de [27] combinent les approches *multithreading* et GPGPU, mais aucun logiciel complet n’est associé à la méthode de décodage logiciel proposée.

Intéressons-nous maintenant aux solutions libres existantes. JasPer [3] est une des premières solutions parues sur le codage et le décodage du JPEG2000. Ce logiciel est sous licence libre. Non seulement JasPer n’est pas caractérisé par de bonnes performances, en

ce qui concerne la rapidité du décodage, mais il n'est plus maintenu par les développeurs. En termes de solutions libres, il a été supplanté par OpenJPEG [1], qui atteint de meilleures performances de décodage et intègre des jeux d'instructions SSE. En revanche, OpenJPEG n'utilise pas de décodage en parallèle. Depuis mai 2015, OpenJPEG constitue un logiciel de référence pour le format JPEG2000⁴.

Nous avons décidé de ne pas recourir au GPGPU, non seulement pour réduire la complexité du décodeur, mais aussi pour ne pas dépendre d'un matériel spécifique. En effet, la nécessité d'utiliser des cartes NVIDIA imposée par l'utilisation de CUDA, qui est un langage de programmation de GPGPU spécifique aux cartes NVIDIA, est une limitation des solutions proposées dans [26] et [31] : même si OpenCL [27] est un langage de programmation de GPGPU prévu pour être indépendant du matériel, il est néanmoins nécessaire, pour atteindre des hauts niveaux de performance, de faire des réglages qui, eux, dépendent du matériel.

Notre choix s'est porté sur une approche *multithreading*, comme dans Kakadu [47], et sur l'utilisation des jeux d'instructions SSE, comme dans OpenJPEG et Kakadu. Le *multithreading* et les jeux d'instructions SSE sont supportés par tous les processeurs pour PC actuels. À la différence d'OpenJPEG et de Kakadu, notre décodeur est spécialisé pour les profils du cinéma numérique du JPEG2000 (norme ISO), ce qui permet de réduire la complexité du code et des structures de données.

4.4.3 Décodage avec pertes

Une autre façon de réduire le temps de décodage est de ne pas complètement décoder le train de bits compressé, c'est-à-dire d'effectuer un décodage avec pertes (on rajoute sciemment des pertes aux pertes inévitablement causées par l'utilisation de transformations non inversibles lors du codage).

Dans [21], une méthode de décompression visuellement sans perte est présentée. Cette méthode se fonde sur le système visuel humain et le calcul d'un seuil de visibilité *Visibility Threshold* (VT) par *code block*, qui permet de déterminer si l'image décompressée est visuellement sans perte. Une caractéristique intéressante de cette méthode est qu'elle ne nécessite pas de connaître la manière dont les images ont été encodées. Pour chaque *code block*, le décodage commence une fois le seuil VT calculé : à chaque passe de codage, une erreur est calculée, qui est comparée au VT ; le décodage du *code block* s'arrête lorsque cette erreur devient inférieure au VT.

Le logiciel Kakadu propose également une option appelée *bitstream truncation*, qui permet de meilleures performances de décompression en ne traitant pas certaines passes de codage. Les auteurs de [11] détaillent la méthode de Kakadu pour le décodage avec pertes. Leur but est de proposer une solution de décodage de DCP pour la prévisualisation. Comme les écrans et les projecteurs usuels ont des résolutions équivalentes au 2K du DCI, ils déconseillent de supprimer un des niveaux de décomposition des DWT, afin de réellement obtenir une image au format 2K. En revanche, ces dispositifs n'affichent les couleurs que sur 8 bits (à comparer aux 12 bits du DCI). Les auteurs de [11] conseillent

4. http://www.jpeg.org/items/20150709_press.html

donc de limiter la précision des 12 bits en la réduisant à 8 bits. Comme la stratégie proposée pour « sauter » des passes de codage n'est pas fondée sur la connaissance a priori du train de bits encodé, elle peut être appliquée à n'importe quelle image compressée.

Cette approche ne nous paraît cependant pas totalement justifiée. En effet, le format des images à encoder est le format TIFF. Chaque composante couleur d'un pixel est codée sur 16 bits, mais seuls les 12 bits de poids fort sont utilisés, les autres bits étant mis à zéro. Avant le traitement par DWT, les données sont donc codées sur 16 bits. Si l'on effectue un codage avec pertes, les DWT sont effectuées en opérations flottantes. Avec plusieurs niveaux de décomposition, les valeurs des coefficients des *code blocks* sont donc totalement modifiées, avant même le traitement des plans de bits par EBCOT. Si l'on se limite à ne décoder les coefficients que sur 8 plans de bits pour avoir une image où chaque canal est codé sur 8 bits, celle-ci sera très dégradée. En particulier, les coefficients d'ondelettes des *code blocks* de la sous-bande de plus faible fréquence (LL) correspondent généralement aux valeurs les plus élevées et sont codés sur 14 plans de bits.

Nous avons choisi d'implémenter une solution qui combine le *multithreading* et les instructions SSE, ainsi que le saut de passes de codage des plans de bits les moins significatifs sur les coefficients d'ondelettes des *code blocks*. Cette technique de décodage est un composant clé de notre système de projection alternatif.

4.5 Description du décodeur proposé

Nous avons écrit un décodeur JPEG2000 spécifique au cinéma numérique, dénommé J2K-libavcodec. Ce décodeur est écrit dans les logiciels multimédia libav et ffmpeg. Il s'agit bien de deux logiciels différents, car il y a eu une scission de la communauté des développeurs. Il n'existait à l'origine que ffmpeg, mais une partie des développeurs a choisi de bifurquer vers une nouvelle instance appelée libav. Les deux logiciels possèdent les mêmes fonctions et les mêmes bibliothèques (libavformat et libavcodec). Nous avons choisi de travailler dans libav, car le code publié dans libav est répercuté dans ffmpeg, ce qui n'est pas toujours le cas dans l'autre sens. Le code du décodeur JPEG2000 sera donc publié dans les deux logiciels. Pour simplifier, nous parlerons uniquement, dorénavant, de la bibliothèque libavcodec, qui est commune aux deux projets. Le code développé dans libavcodec est publié à la fois dans libav et dans ffmpeg.

Comme nous l'avons déjà dit, il existe d'autres décodeurs JPEG2000 libres. En particulier, OpenJPEG est le logiciel libre de référence pour le décodage JPEG2000, qui est plus dédié au décodage des images qu'à celui des vidéos. Il semble intéressant de comparer les performances de ce logiciel de décodage non spécifique au cinéma numérique à celles de notre décodeur. Il se trouve que OpenJPEG est également disponible dans libavcodec, pour peu que l'édition de liens soit effectuée, ce qui n'est pas nécessaire pour J2K-libavcodec, puisque son code est inclus dans cette bibliothèque.

Grâce à la spécialisation de notre décodeur, qui est dédié au JPEG2000 pour le cinéma, nous avons pu développer des structures de données et un code plus simples, qui évitent ainsi les structures complexes d'OpenJPEG nécessaires à l'implémentation de toutes les options du JPEG2000.

Notre décodeur est bien sûr de type « intra-trame », le DCI ayant proscrit toute compression temporelle à l'intérieur de séquences de trames. Nous avons mis en place une exécution parallèle du décodage à l'intérieur de chaque trame, en utilisant les primitives de *threading* de libavcodec. La transformation en ondelettes discrète inverse, ou *Inverse Discrete Wavelet Transform* (IDWT), et la transformation en couleur inverse, ou *Inverse Multi-Component Transform* (IMCT), sont optimisées avec des instructions SSE.

Nous savons par avance que toutes les images auront la même dimension, le même nombre de composantes, le même nombre de niveaux de décomposition, ... Pour augmenter la rapidité de traitement, nous ne traitons donc qu'une seule fois certaines parties des en-têtes.

Le code de notre décodeur est public et librement accessible dans le répertoire libavcodec de libavcodev (cf. figure 4.6). Pour accéder au code publié, il suffit de taper :

```
1 git clone git://git.libav.org/libav.git
```

Toutes les optimisations n'ont pas encore été publiées dans libavcodec, elles sont toutefois disponibles sur nos dépôts git :

```
1 git clone git://git.utopialab.tetaneutral.net/libav.git
```

```
libavcodec
├─ jpeg2000.c.....Initialisations et traitement des en-têtes
├─ jpeg2000.h.... Déclaration des structures de données et des constantes
├─ jpeg2000dec.c.....Fonctions de décodage
├─ jpeg2000dwt.c.....Traitement de l'IDWT
├─ jpeg2000dwt.h..... Déclaration des fonctions de l'IDWT
├─ jpeg2000dsp.c.....Fonctions à optimiser
├─ jpeg2000dsp.h..... Déclaration des fonctions à optimiser
├─ mqc.c.....Implémentation du codeur MQ
├─ mqc.h.....Structures de données du codeur MQ
├─ mqcdec.c..... Partie décodage du codeur MQ
```

FIGURE 4.6 – Organisation du code source du décodeur JPEG2000 dans libavcodec.

4.6 Évaluation

4.6.1 Évaluation du décodeur

Les tests du décodeur J2K-libavcodec ont été effectués sur deux machines :

i7 – Un portable équipé d'un processeur Intel core i7-2820QM cadencé à 2,30 MHz et d'un processeur à 4 cœurs, pour un total de 8 *threads* de type *Simultaneous MultiThreading* (SMT), appelés *hyperthreads* dans la terminologie Intel.

BiXeon – Une station de travail équipée de deux processeurs Xeon E5-2620 cadencés à 2,0 GHz. Chaque processeur possède 6 cœurs, pour un total de 24 SMT *threads*.

Dans tous les tests, le mode *hyperthreading* est activé. Les tests sont réalisés pour différents nombres de SMT *threads*, et à différentes résolutions.

Le DCP testé est le film-annonce de *Moonrise Kingdom*. Tous les résultats sont relatifs au dix premières secondes (240 trames). La taille des images est 1998×1080 . Le DCP a été compressé à un débit binaire de 128 Mb/s. Lors du codage, $N = 5$ DWT ont été appliquées aux images. Le niveau de résolution le plus élevé traité lors du décodage est $N = 5$ (format 2K, images 1998×1080) ou $N - 1 = 4$ (format 1K, images 999×540).

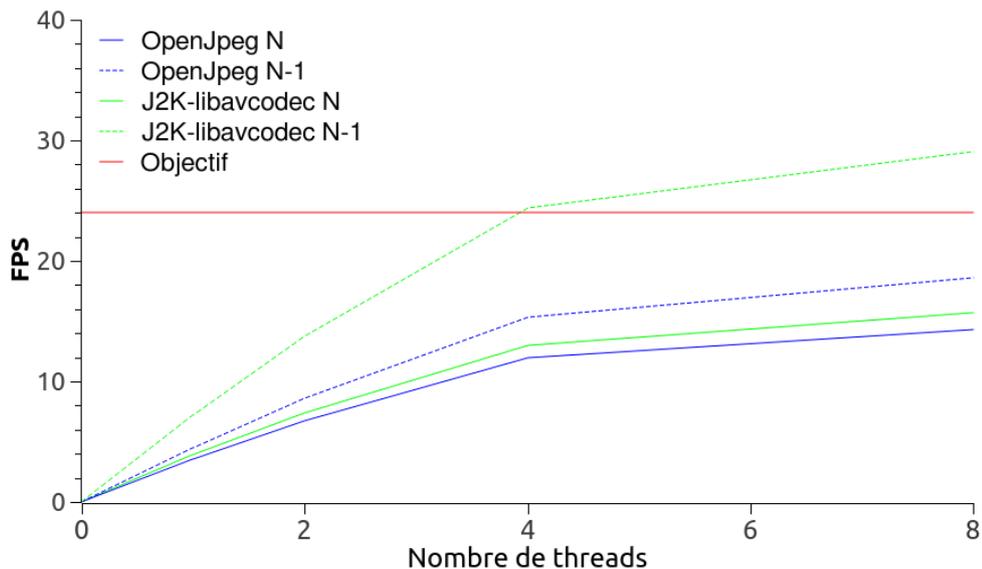


FIGURE 4.7 – Débits de décodage (en fps) de OpenJPEG et J2K-libavcodec sur la machine i7, en fonction du nombre de *threads*, pour les niveaux de résolution $N = 5$ et $N - 1 = 4$ (tests effectués sur un échantillon du film-annonce de *Moonrise Kingdom*).

La figure 4.7 permet de comparer les performances des décodeurs OpenJPEG et J2K-libavcodec sur la machine i7. Comme le processeur de cette machine comporte 8 SMT *threads*, les mesures de performances de décodage sont réalisées jusqu'à 8 *threads* de libavcodec.

Au format 2K, ni OpenJPEG ni J2K-libavcodec n'atteignent l'objectif de 24 fps (*frames per second*). Néanmoins, notre décodeur atteint de meilleures performances. Au format 1K, J2K-libavcodec atteint 24 fps avec 4 *threads*, contrairement à OpenJPEG.

Les résultats des mêmes tests effectués sur la machine BiXeon sont présentés sur la figure 4.8. Tous les décodeurs, quel que soit le niveau de résolution, atteignent le débit de 24 fps. Cependant, J2K-libavcodec atteint l'objectif avec 12 *threads*, alors que OpenJPEG l'atteint avec 16 *threads*.

Bien sûr, les performances atteintes sont meilleures pour le format 1K que pour le format 2K. L'image à décoder au format 1K est quatre fois plus petite qu'au format 2K,

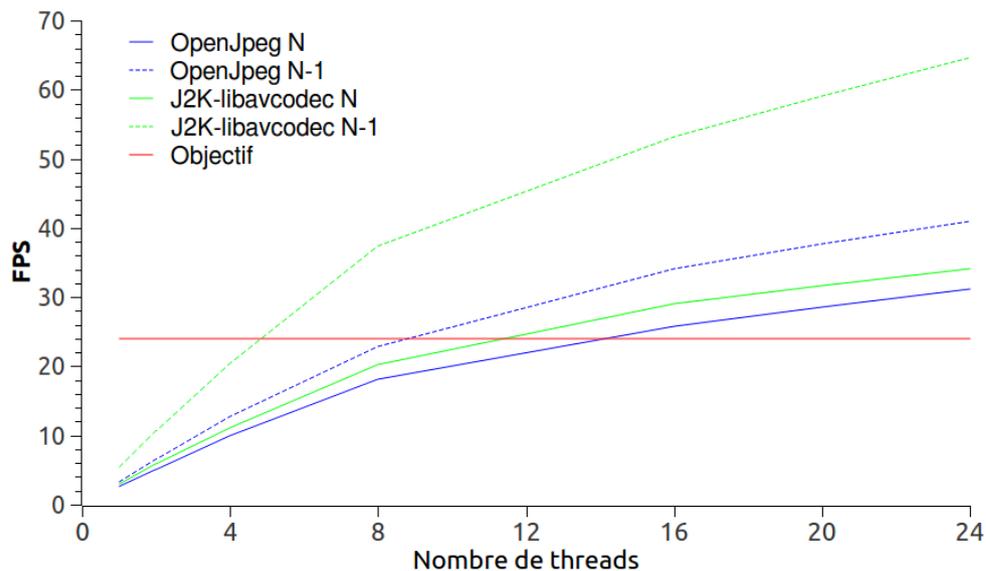


FIGURE 4.8 – Débits de décodage (en fps) de OpenJPEG et J2K-libavcodec sur la machine BiXeon, en fonction du nombre de *threads*, pour les niveaux de résolution $N = 5$ et $N - 1 = 4$ (tests effectués sur un échantillon du film-annonce de *Moonrise Kingdom*).

donc il y a quatre moins de *code blocks* à décodé. Néanmoins, on ne peut pas espérer un gain égal à 4, car les *code blocks* non décodés sont ceux des hautes fréquences (en l'occurrence, ceux des sous-bandes 5HL, 5LH et 5HH). Or, EBCOT encode les *code blocks* des hautes fréquences avec moins de bits que ceux des basses fréquences (les taux de compression sont plus élevés dans les basses fréquences).

Le gain atteint par notre décodeur, lorsque nous passons du format 2K au format 1K, est plus élevé que le gain atteint par OpenJPEG. Avec OpenJPEG, ce gain est de l'ordre de 1,5, alors que nous observons avec J2K-libavcodec un gain de l'ordre de 2. Cette différence est due à l'usage de la mémoire et à la gestion du cache CPU : nous utilisons des structures plus petites pour décodé le flux de données, ce qui permet de limiter le nombre de chargements de données dans les caches du CPU.

La table 4.2 montre comment le débit de décodage (en fps) varie, pour des films compressés à différents débits binaires. Les films choisis sont des films-annonces. Le décodage est testé sur les 240 premières trames. Comme cela était prévisible, les films les plus compressés (débit binaire faible) sont ceux pour lesquels le décodage est le plus rapide. Pour *Spring Breakers* et *Django Unchained*, nous obtenons quasiment le même débit de décodage, alors que les débits binaires sont différents. Il se trouve que les 240 premières trames de *Spring Breakers* comportent beaucoup d'images noires. Or, le décodage d'images noires est très rapide pour un décodeur JPEG2000.

TABLE 4.2 – Débits de décodage de J2K-libavcodec sur la machine BiXeon, en fonction du nombre de *threads* (12 ou 24), obtenus sur des échantillons de films-annonces compressés à différents débits binaires.

Film-annonce	Débit binaire (Mb/s)	12 <i>threads</i> (fps)	24 <i>threads</i> (fps)
<i>Spring Breakers</i>	201	19,05	24,62
<i>Django Unchained</i>	151	19,58	24,64
<i>Moonrise Kingdom</i>	128	27,12	33,76
<i>Jeux d'été</i>	79	30,34	38,28

4.6.2 Évaluation du décodage avec pertes

La table 4.3 présente les performances du décodage lorsque plusieurs passes de codage sont sautées. Les tests ont été réalisés sur la machine BiXeon avec 24 *threads*. En sautant 5 passes, pour les niveaux de résolution $N = 5$ et $N - 1 = 4$, les gains de performances sont remarquables (respectivement 40 % et 50 %). Les gains de performances sont encore plus élevés en sautant 20 passes de codage.

TABLE 4.3 – Débits de décodage (en fps) de J2K-libavcodec sur la machine BiXeon avec 24 *threads*, pour les niveaux de résolution $N = 5$ et $N - 1 = 4$, obtenus sur un échantillon du film-annonce de *Moonrise Kingdom* en sautant des passes de codage (0, 5 ou 20).

Nombre de passes sautées	$N = 5$ niveaux de résolution	$N - 1 = 4$ niveaux de résolution
0	33,21	63,16
5	46,08	94,71
20	61,26	204,08

Si les gains des débits de décodage obtenus en sautant des passes de codage sont très élevés, encore faut-il que cela n'affecte pas trop la qualité d'image. Nous avons donc effectué des tests de mesure de qualité d'image sur le même échantillon que précédemment, à savoir les 240 premières trames du film-annonce de *Moonrise Kingdom*. Pour pouvoir mesurer des valeurs de *Peak Signal to Noise Ratio* (PSNR), il nous faut une référence. Nous utilisons bien sûr le même échantillon décodé au format 2K.

Pour tester les performances du décodage avec pertes, nous ne décodons pas les dernières passes de codage d'EBCOT. Nous ignorons ainsi les bits les moins significatifs des coefficients des *code blocks*. Sur la figure 4.9, nous présentons les valeurs de PSNR des 240 trames de l'échantillon, en sautant soit 5 passes, soit 20 passes de codage. Nous présentons également les valeurs de PSNR obtenues avec une décompression au format 1K, suivie d'une interpolation bicubique pour atteindre le format 2K.

L'évolution du PSNR sur le graphique de la figure 4.9 nous permet de repérer les différentes séquences de cet échantillon de 240 trames : images noires (pic des premières trames), affichage du logo du Festival de Cannes (jusqu'à la trame 55) puis du logo de Studio Canal (jusqu'à la trame 77), images noires (pic entre les trames 78 et 82), et enfin plusieurs séquences du film. Des sous-titres sont affichés entre les trames 148 et 176, puis à partir de la trame 202 : la courbe verte décroît nettement pendant ces deux séquences, ce qui semble indiquer que le format 1K interpolé « bave » au bord des sous-titres (le même phénomène se produit pendant l'affichage des deux logos, qui possèdent eux aussi des bords saillants).

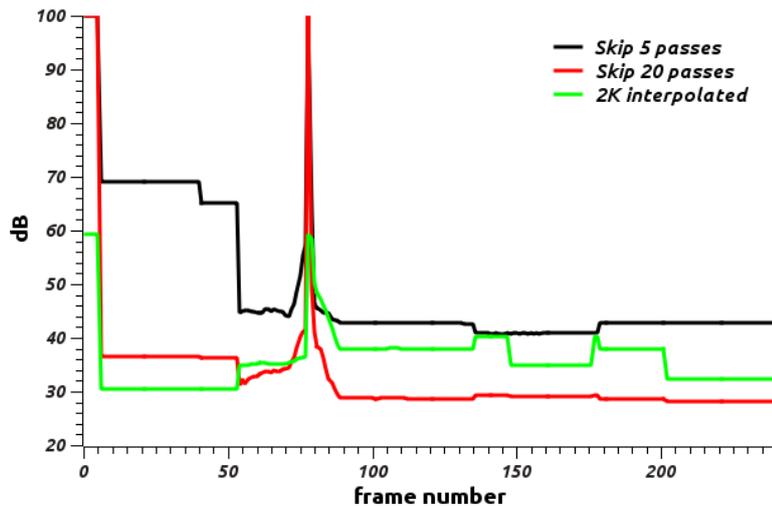


FIGURE 4.9 – Évolution du PSNR pour les 240 premières trames du film-annonce de *Moonrise Kingdom*, pour trois variantes de notre décodeur J2K-libavcodec : avec 5 passes sautées (en noir), avec 20 passes sautées (en rouge), ou en version 1K interpolé (en vert). Les pics situés au début de l'échantillon, puis autour de la trame 80, correspondent à des images noires.

L'inspection visuelle de la vidéo, qui est forcément subjective, nous permet d'observer que la qualité d'image est « acceptable » pour la projection lorsque 5 passes sont sautées. Lorsque 20 passes sont sautées, la qualité d'image n'est plus acceptable pour la projection, mais reste acceptable pour la prévisualisation de DCP.

Dans [21], les qualités d'image entre décodage sans perte et décodage avec pertes sont comparées par une évaluation perceptuelle, mais également à l'aide de deux mesures de similarité entre images numériques : PSNR et *Structural SIMilarity* (SSIM). Notre étude nous permet de confirmer les résultats de cet article, à savoir que le décodage visuellement sans perte est avantageux en termes de performances de décodage.

4.7 Bilan

Nous avons implémenté un décodeur JPEG2000 dédié au cinéma numérique dans la bibliothèque libavcodec. Nous avons comparé les performances de notre décodeur à celles du décodeur de référence OpenJPEG, qui est le plus rapide des décodeurs JPEG2000 libres.

Nous avons montré que notre décodeur était plus rapide que OpenJPEG, surtout quand le niveau de résolution des fréquences les plus élevées n'est pas décodé. De plus, nous pouvons atteindre d'importants gains de performances si nous n'effectuons pas toutes les passes de codage d'EBCOT, tout en garantissant un décodage visuellement sans perte.

La comparaison de notre décodeur avec des décodeurs JPEG2000 « propriétaires » est plus délicate. Le site web officiel de Kakadu fournit bien un certain nombre de résultats, mais une comparaison directe est impossible car l'échantillon utilisé pour les tests n'est pas disponible. Les images testées sont plus petites de 20 % que celles des tests présentés dans ce chapitre, mais le débit binaire (244 Mb/s) est deux fois supérieur à celui de notre échantillon de test (*Moonrise Kingdom*).

Sur une machine avec processeurs Bi-Xeon, Kakadu atteint un débit de décodage de 24,24 fps (35,08 fps avec l'option *speed pack*). À titre de comparaison, nous atteignons un débit de décodage de 34,04 fps sur le même type de machine.

EasyDCP, autre solution « propriétaire », ne rend publique aucune donnée relative à ses performances de décodage.

Le nombre de cœurs CPU étant croissant sur les machines actuelles (Intel vient de lancer un processeur à 8 cœurs), nous pensons pouvoir éviter l'utilisation de GPGPU pour le décodage JPEG2000 avec des taux de compression élevés.

Nous nous sommes intéressés dans ce chapitre au décodage d'un DCP, vu comme un flux binaire d'images JPEG2000. Or, chaque image de ce flux est chiffrée. Il nous faut donc maintenant aborder la question du déchiffrement des images, et plus largement celle de la sécurité dans le cinéma numérique.

Chapitre 5

Contributions à la sécurisation des contenus

5.1 Introduction

La grande majorité des DCP sont chiffrés. Le besoin de pouvoir lire des DCP chiffrés est donc incontournable. Le chiffrement permet de lutter contre le vol de contenu. Il est également utile pour le distributeur, car cela lui permet de contrôler la distribution des films dans les salles : il ne génère une clé de déchiffrement que pour une période donnée.

La sécurité est une composante importante du cinéma numérique. Par sécurité, on entend la lutte contre le piratage des contenus. La spécification de la sécurité constitue la majeure partie de la DCSS (60 pages sur 150), qui compte huit autres parties dans les 90 pages restantes. Une partie importante du coût des systèmes de projection est d'ailleurs due à l'implémentation de ces exigences de sécurité [4].

Un enjeu majeur de cette thèse consiste à vérifier que l'on puisse chiffrer et déchiffrer les DCP, de même que les messages de déchiffrement (KDM) associés, en n'utilisant que du logiciel libre. Les méthodes de chiffrement utilisent de nombreux protocoles et algorithmes (AES, RSA, X.509). Peut-on implémenter ces algorithmes sans passer par des organismes gérant ces protocoles ?

5.2 La sécurité dans le cinéma numérique

La spécification de la DCSS répertorie six types de niveaux de sécurité [9] :

Chiffrement de la copie numérique – Chaque bobine virtuelle associée à une piste vidéo, son ou sous-titres d'une œuvre numérique est chiffrée avec une clé symétrique issue du standard AES [34]. Le chiffrement symétrique doit être utilisé dans le mode *Cipher Block Chaining* (CBC) avec une clé de 128 bits [42], chaque trame d'un fichier MXF étant encodée avec ce chiffrement. Le chiffrement AES est détaillé dans le paragraphe 5.2.1. Notons que le chiffrement du contenu n'est pas obligatoire, c'est-à-dire qu'un DCP peut être diffusé « en clair ».

Transport de clé – Les clés de déchiffrement AES ayant servi au chiffrement des contenus d’un DCP sont contenues dans un KDM, qui est un fichier XML contenant l’ensemble des données nécessaires au déchiffrement. Le fonctionnement des KDM est décrit dans le paragraphe 5.2.3. La partie du KDM qui contient les clés AES du DCP est chiffrée avec un algorithme de chiffrement asymétrique (cf. paragraphe 5.2.2).

Security Manager (SM) – Ce composant du *media block* coordonne les activités liées à la sécurité dans une salle de projection. Il authentifie les autres composants liés à la sécurité, comme le SMS ou le *Media Decryptor*¹, par le biais de certificats numériques. Le fichier KDM contient, entre autres, la liste des composants autorisés à jouer une œuvre numérique, appelée *Trusted Device List* (TDL). Le SM vérifie si les équipements de projection installés sont conformes à la TDL. Le SM est également chargé d’établir des sessions sécurisées de type *Transport Layer Security* (TLS) pour assurer des communications sécurisées au travers du réseau local du cinéma.

Sécurité physique – Chaque composant de sécurité est implémenté sur du matériel sécurisé, c’est-à-dire sur un matériel qui protège physiquement l’accès aux données sensibles. Par exemple, la détection de l’ouverture du projecteur déclenche une alarme. Dans ce domaine, la spécification du DCI demande que le matériel soit certifié au standard *Federal Information Processing Standards* (FIPS) 140-2 de niveau 3².

Traces sécurisées – Afin de rendre flexibles les opérations des exploitants, une approche *Control Lightly/Audit Tightly* est adoptée ([10], page 136). Au lieu d’utiliser des mesures de protection complexes, comme les *Digital Rights Management* (DRM), pour contrôler comment le contenu est utilisé, les propriétaires de droits d’auteur spécifient une période pendant laquelle l’œuvre peut être jouée. Les exploitants doivent fournir des traces (*logs*) montrant l’usage du contenu. Ces traces doivent être sécurisées afin de garantir leur intégrité et leur authenticité.

Marquage des films – Appelé en anglais *forensic watermarking*, le marquage consiste à ajouter des messages invisibles à l’œil humain ou inaudibles à l’oreille humaine. Le but est de pouvoir identifier quand et où un film a été enregistré illégalement par une caméra dans une salle de cinéma, afin de décourager le piratage de films. Le marquage doit encore être détectable après captation du film par le pirate. Comme cela est souligné à la page 139 de [30], il s’agit du sujet le plus flou de la spécification du DCI. À ce jour, il n’existe aucun standard sur le marquage des films. Dans [28], la lutte contre le piratage va encore plus loin : un modèle mathématique est proposé pour détecter la position de la caméra du pirate dans la salle de projection, en fonction de la distorsion des images due à l’angle sous lequel la caméra voit l’écran.

1. Le *Media Decryptor* transforme des images ou des sons chiffrés dans leur forme claire originale (voir le paragraphe 9.3.3.2 de [16] pour une liste exhaustive de ces entités).

2. Pour une présentation du standard de sécurité FIPS 140-2 : [30], pages 151-154, et http://en.wikipedia.org/wiki/FIPS_140-2

Les moyens de sécurisation visant à lutter contre le piratage vont de la génération de la copie numérique à la projection en salle. La spécification du DCI et les standards de la SMPTE ne traitent pas de la sécurité entre la production et la distribution de la copie numérique. Le DCI considère (cf. paragraphe 8.4.3.1 de la DCSS) que, lors de l'étalonnage du *master* numérique, il n'est pas nécessaire de chiffrer la liaison entre le *Media Block* et le projecteur, car il s'agit d'un environnement fiable (*trusted environment*). Cependant, l'étude menée par [13] montre que 75 % des films que l'on retrouve sur les réseaux « pair à pair » illégaux proviennent de personnes travaillant dans la production ou la distribution cinématographique, qui copient et redistribuent illégalement les films.

5.2.1 Chiffrement AES

Chaque trame d'un fichier MXF étant chiffrée avec l'algorithme *Advanced Encryption Standard* (AES), une présentation de ce mode de chiffrement s'impose.

AES est un algorithme de chiffrement « symétrique » : la même clé est utilisée pour le chiffrement et le déchiffrement. AES divise les données à chiffrer en blocs de 128 bits (16 octets) et utilise ces blocs en entrée de l'algorithme. La clé de chiffrement AES peut avoir une taille de 128, 192 ou 256 bits. AES applique sur les blocs de données de multiples transformations combinées avec la clé de chiffrement. Ces itérations sont appliquées sur 10, 12 ou 14 cycles correspondant à des clés de 128, 192 ou 256 bits, respectivement.

Le mode opératoire décrit comment les blocs chiffrés seront organisés dans le fichier de chiffrement final. En effet, il ne s'agit pas d'une simple concaténation des blocs chiffrés, car le résultat final serait trop vulnérable (similarité entre blocs contenant les mêmes données). Il existe plusieurs modes opératoires (ECB, CBC, OFB, ...) correspondant à différents niveaux de sécurité.

Pour le cinéma numérique, le mode opératoire CBC a été choisi par la SMPTE, conjointement au chiffrement AES. Une opération binaire de type « ou exclusif » est effectuée entre le premier bloc non encore chiffré et le dernier bloc chiffré. Le résultat de cette opération est ensuite chiffré en utilisant la clé. Ainsi, le chiffrement d'un bloc dépend du bloc précédent. Un « vecteur d'initialisation », généré pseudo-aléatoirement et stocké dans le fichier MXF chiffré, est utilisé pour le premier bloc (cf. figure 5.1).

5.2.2 Chiffrement RSA

Le chiffrement *Rivest-Shamir-Adleman* (RSA) a été inventé en 1977 [35]. L'algorithme RSA fait partie des algorithmes de chiffrement à clé publique, encore appelé chiffrement « asymétrique » car la clé de chiffrement n'est pas la même que celle utilisée pour le déchiffrement. Deux clés sont nécessaires, une dite « publique » pour le chiffrement, une autre dite « privée » pour le déchiffrement.

La SMPTE [45] impose que l'on utilise le schéma de remplissage *Optimal Asymmetric Encryption Padding* (OAEP). Le remplissage est le processus qui consiste à organiser le flux de données avant chiffrement. Le remplissage OAEP est le remplissage actuellement le plus sûr, le couple RSA/OAEP étant la méthode de chiffrement asymétrique recommandée par l'ISO.

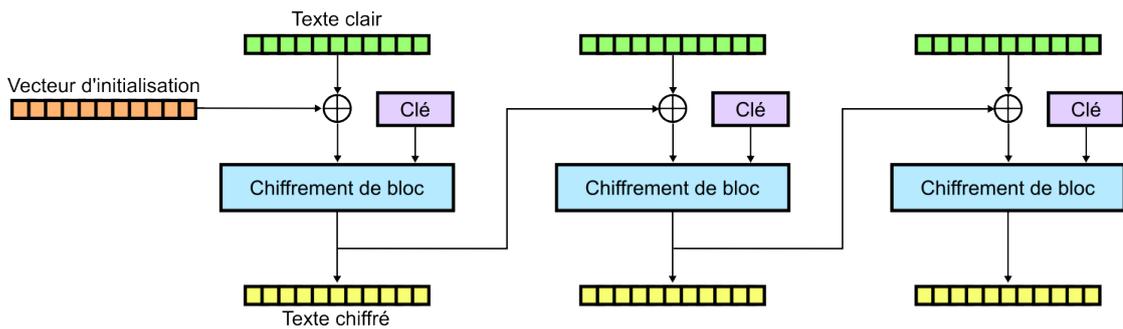


FIGURE 5.1 – Mode opératoire CBC de chiffrement. Source wikipédia https://commons.wikimedia.org/wiki/File:Schema_cbc.png.

Pour le chiffrement RSA, la SMPTE [45] exige que l'implémentation corresponde au standard *Public Key Cryptographic Standards (PKCS)#1* [25]. Le DCI a choisi d'utiliser des algorithmes de chiffrement asymétrique de type RSA, avec une clé publique de 2048 bits.

5.2.3 Fichier KDM

Le but du KDM est de transporter de manière sûre des messages chiffrés entre équipements numériques. C'est un fichier XML qui contient principalement trois types d'informations :

- Les clés AES du DCP associé.
- Les paramètres pour l'utilisation de la clé, comme la période de validité de la clé.
- La TDL, qui identifie les équipements autorisés à utiliser les clés de chiffrement AES.

L'accès à toutes les informations du KDM nécessite la clé privée du destinataire, car les parties essentielles du KDM sont chiffrées avec l'algorithme RSA. La figure 5.2 donne une vue globale d'un KDM.

Chaque KDM est créé spécifiquement pour chaque cinéma, et si le cinéma possède plusieurs salles, un KDM est créé pour chaque projecteur. C'est le distributeur qui génère le KDM pour chaque projecteur, pour une période donnée (généralement une semaine).

Ce mécanisme allège la distribution des DCP, car un DCP est chiffré une seule fois, puis le même DCP chiffré peut être distribué à tous les cinémas. Seuls les équipements de projection ayant reçu un KDM pourront déchiffrer le DCP. Le KDM est un fichier ne pesant que quelques ko. Il est généralement distribué par courriel, contrairement au DCP chiffré qui pèse plusieurs Go et doit être distribué par des moyens appropriés (cf. chapitre 6).

La relation unique entre le KDM et le projecteur est faite en chiffrant dans le KDM les clés AES du DCP avec la clé publique du projecteur. Le lien entre les clés AES stockées dans le KDM et le DCP est fait à travers les fichiers CPL. Chaque fichier KDM contient l'identifiant (de forme UUID) qui permet de faire le lien avec la CPL du DCP

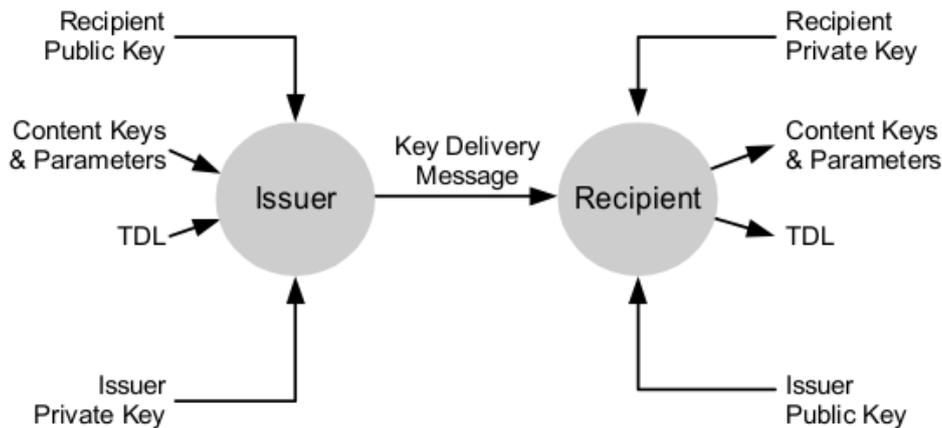


FIGURE 5.2 – Vue globale du fonctionnement d’un KDM (source [41]).

associé. Le lien entre la clé AES et le fichier MXF chiffré se fait au travers d’un identifiant présent à la fois dans la CPL et dans le KDM. Ces liens sont illustrés sur la figure 5.3.

Un KDM est une instance du format *Extra-Theater Messages* (ETM) défini par la SMPTE [45]. Ce format sert à la communication entre équipements du cinéma numérique. Il contient notamment les informations liées à la signature du document, afin de s’assurer de son origine. Un KDM est donc à la fois chiffré de manière asymétrique pour son contenu le plus sensible (i.e. les clés AES du DCP), et également signé afin de garantir l’origine du KDM.

La signature consiste à utiliser des certificats numériques. La SMPTE [40] utilise le format X.509v3, auquel elle ajoute des contraintes. Ce format est très répandu, par exemple pour l’accès aux sites Internet sécurisés (SSL/TLS). Les contraintes imposées par la SMPTE visent à utiliser un sous-ensemble du format X.509v3 qui garantisse l’interopérabilité et un haut niveau de sécurité. Le format X.509 lie une clé publique (de type RSA) à une entité (personne, entreprise, ...) au travers de la génération d’un certificat.

Il existe des sites web qui centralisent les certificats, notamment le site arcene du CNC et celui de la société Doremi, qui fabrique plus de 90 % des serveurs de projection en France.

5.3 Chiffrement des DCP

Dans les différents items de sécurité définis par le DCI, nous avons traité le cas de la gestion de KDM. Notre contribution est double : d’une part, la génération de KDM pour un équipement de projection ; d’autre part, le déchiffrement d’un DCP.

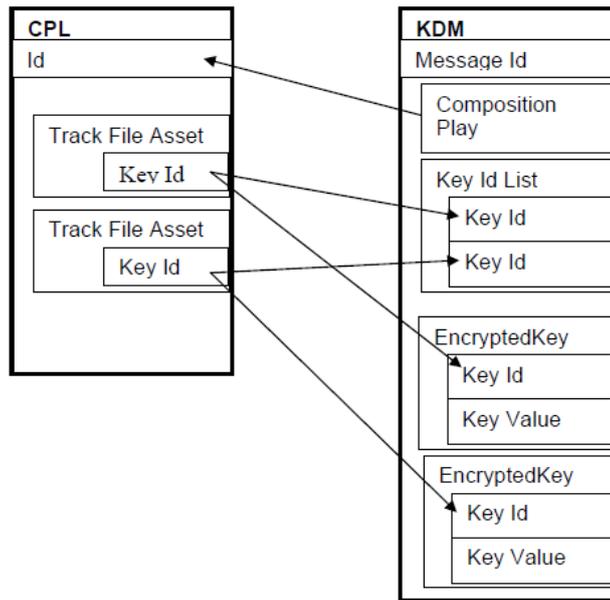


FIGURE 5.3 – Liens entre KDM et CPL (source [41]).

5.3.1 Génération de KDM

Pour générer un fichier KDM permettant de déchiffrer un contenu pour un équipement de projection, il nous faut :

- La clé AES ayant permis le chiffrement du DCP.
- Les *KeyId* du fichier CPL.
- La clé publique (également appelée « certificat ») de l'équipement pour lequel il faut générer le KDM.
- La liste TDL de l'équipement de projection.

Il existe de nombreuses solutions pour générer des DCP chiffrés, comme l'outil *cinAsset* de Doremi, mais ces solutions sont « propriétaires ». Les DCP chiffrés sont généralement réalisés dans les laboratoires de post-production. Il existe aussi des solutions libres de création de DCP comme *OpenDCP* ou *DCPoMatic*.

La génération de KDM est gérée par les distributeurs. Il existe des outils, comme ceux de *Cinego*, qui génèrent des KDM. Généralement, le fichier DCP chiffré n'est pas fourni au distributeur. Les usages actuels portent plutôt sur l'utilisation de *Distribution KDM* (DKDM). Ce fichier, qui respecte le format d'un KDM, contient les informations de la CPL et les clés de chiffrement du DCP. Le DKDM n'est pas un format standard, i.e. il n'est défini ni par le DCI ni par la SMPTE.

Dans notre cas, nous voulons créer un DCP chiffré pour le lecteur VLC et le fichier KDM associé. Nous n'avons donc pas à nous soucier des DKDM, car le DCP et la génération de KDM sont réalisés sur la même machine.

Nous devons générer une clé publique/privée sur la machine qui lit des DCP sous VLC. Pour cela, nous utilisons un outil fourni par *Digital Cinema Tools Distribution*³, qui permet de gérer les certificats et les KDM. Avec cet outil, nous générons la clé RSA privée, ainsi que le certificat pour la génération de KDM. Pour que VLC puisse prendre en compte la clé privée, nous la déplaçons dans un répertoire de configuration de VLC (répertoire `.config/vlc/` du compte utilisateur de VLC). Le certificat est un message encodé au format texte, qui contient les données au format X.509, dont la clé publique d'un équipement de projection permettant de chiffrer un DCP (cf. figures 5.4 et 5.5).

```

-----BEGIN CERTIFICATE-----
MIIEezCCA20GAWIBAgIBBzANBgkqhkiG9w0BAQsFADCBiJEUMBIGA1UECgwLZXhh
bXBsZS5vcmcxPDASBgNVBAsMC2V4YW1wbGUub3JnMTUwMwYDVQDDCwuc21wdGUT
NDMwLTIuSU5URVJNRURJQVRFLk5PVF9GT1JfUFJPRFVDE1PTjE1MCMGA1UELHMc
OT1Vb29LTE1WUWtwK1RzV0YxZ01FMTJpS3RnPTAeFw0xNTA4MjUxNjA2MjBaFw0y
NTA4MjUxNjA2MjBaMIGEMRQwEgYDVQKDA1eGFtcGx1Lm9yZzEUMBIGA1UECwWL
ZXhhbXBsZS5vcmcxLzAtBgNVBAMMJkNTLnNtcHRILTRzMC0yLkxkFQUyUtk9UXOZP
U19QUk9EVUNUSU9OMSUwIwYDVQQwExxxncE9ERks4Nnh1UFVFRMct2K1IzdXhUbjhP
YzA9MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAvKAr/r+OvyJr1RSh
tDXYRvt0S7RVoAeeBaPyOYX/k7Yz85hfq1q196ZuT/bTH70xonZWSNcsFRHMqYE
if84fUInQcaQ3jVoKFD6M6ZwTKD70GqzJLx/DWxr3MNexEr7Ih6TcEHkI18WZrK
Q6qvD8oNte603jWgJeFoQldt8H7+BYA/X7pGBLR5X0iNOX1e7dB8WhCeIw5n5UL
W2+Jp2I8AK77sAAmuSowUUtAbkxwPtWya24e90cHPXrEUmqLd9xHnXhc8FIImFJx
y3mMJtaVRyXCuF3x1X07M2n/BORrWYi5erxFRkPf8/YgyvK8y7HSvlyV8qcFRQD0
6p1KQwIDAQABo4HvMIHsMAwGA1UdEwEB/wQCMAAwCwYDVROPAQDAgWgMBOGA1Ud
DgQWBBSck4MUrzrF49RDT6/5He7F0fw5zTCBrwYDVR0jBIGnMIGkgBT31SigosxV
CSn50pYXWAwTXaIq2KGBiKSBhTCBjEUMBIGA1UECgwLZXhhbXBsZS5vcmcxPDAS
BgNVBAsMC2V4YW1wbGUub3JnMS0wKwYDVQDDCQuC21wdGUTNDMwLTIuUk9PVC50
T1RfRk9SX1BSTORVQ1RJT04xJTAjBgNVBC4THGpBdG5PNjdsUStGUTk0TOhodnFK
TD15dlFTMD2CAQYwDQYJKoZIhvcNAQELBQADggEBAJrGt9ed0+30r6yHC14j1wn
6dvJ0ugyk7Qk+vKPCtcg5zrwbZqOmtGM/EtwsF6mdixFtldsJZK2ak9wjGByxy4G
44yb2z5hbDs1cEdJmhVtQ3i5+aKyJidHtVBiKJ8kijK18cXzYBx530bA8WommGV
rH+0rJ4JrTa32bJ/Di/Yhh31YnH50wvf05CmxNO+AibbmY1A++PcpMZBtTDJ/XPZ
LUJ7d4G2V4bszLzzy/2sP1ygEHR3FUbWg4UGp00pkAiOMVCas7zSm4gygcdhFuhlz
riTibpiFvGGMZwxS3YfWJGGIZt/ziKdXowJ0vCACARxPkdrTb/K5kXFmsFpkg=
-----END CERTIFICATE-----

```

FIGURE 5.4 – Exemple de fichier de certificat non décodé.

Une solution très simple aurait consisté à inclure dans le code de VLC la clé privée et le certificat, et à diffuser ensuite le certificat de la clé publique pour la génération de KDM. Il aurait donc fallu générer un seul fichier KDM permettant de lire un DCP chiffré sur toutes les machines équipées du lecteur VLC. Mais cela aurait impliqué une grave faille de sécurité, car la clé privée aurait été exposée publiquement (dans le code source de VLC).

Pour augmenter le niveau de sécurité, nous préférons générer une clé privée et un certificat pour chaque installation de VLC. La génération de la clé privée et celle du certificat se font indépendamment de l'installation de VLC. Il suffit ensuite de faire publier le certificat dans une base de données, par exemple dans la base arcene du CNC.

Pour générer un DCP chiffré, nous avons utilisé le logiciel DCP-o-matic, qui permet de générer des DCP mais aussi le KDM associé au DCP pour un équipement de projection.

3. https://github.com/wolfgangw/digital_cinema_tools_distribution/wiki

```

Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number: 7 (0x7)
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: O=example.org, OU=example.org,
           Validity
           Not Before: Aug 25 16:06:20 2015 GMT
           Not After : Aug 20 16:06:20 2025 GMT
    Subject: O=example.org, OU=example.org,
           Subject Public Key Info:

    Public Key Algorithm: rsaEncryption
    Public-Key: (2048 bit)
    Modulus:
      00:bc:a0:2b:fe:bf:8e:bf:22:6b:d5:14:a1:b4:35:
      d8:46:fb:74:4b:b4:55:d0:07:9e:05:a3:f2:d1:85:
      ff:93:b6:33:f3:98:5f:ab:5a:a5:f7:a6:6e:4f:f6:
      d3:1f:b3:b1:a2:76:56:f1:23:5c:b0:54:47:32:a6:
      04:89:ff:38:7d:42:27:43:47:1a:43:78:d5:a0:a1:
      43:e8:ce:99:c1:32:83:ef:41:aa:cc:92:f1:fc:35:
      b1:af:73:0d:7b:11:2b:ec:88:7a:4d:c1:07:90:89:
      7c:59:9a:ca:43:aa:af:0f:ca:0d:b5:ee:8e:de:35:
      a0:25:e1:68:42:57:6d:f0:7e:fe:05:80:3f:5f:ba:
      46:06:54:79:5c:e8:8d:39:7d:5e:ed:d0:5e:f1:68:
      42:78:8c:39:9f:95:0b:5b:6f:89:a7:62:3c:00:ae:
      fb:b0:00:26:b9:2a:30:51:4b:5a:6e:4c:70:3e:d5:
      b2:6b:6e:1e:f4:e7:07:3d:7a:c4:52:6a:a0:2d:df:
      71:1e:75:e1:73:c1:48:98:52:71:cb:79:8c:26:d6:
      95:47:25:c2:b8:5d:f1:d5:73:bb:33:69:ff:04:e4:
      6b:59:88:b9:7a:bc:45:46:43:df:f3:f6:06:ca:f2:
      bc:cb:b1:d2:be:5c:95:f2:a7:05:45:00:ce:ea:99:
      4a:43
    Exponent: 65537 (0x10001)

```

FIGURE 5.5 – Version décodée du certificat de la figure 5.4 : nous retrouvons la clé publique sur 2048 bits (Public-Key : (2048 bit)).

La SMPTE demande d'utiliser une liste TDL pour générer un KDM (cf. figure 5.2), mais il s'avère que la TDL n'est pas nécessaire pour générer un KDM [41]. Effectivement, de nombreux outils très utilisés dans l'exploitation cinématographique, comme ceux de Cinego, n'utilisent pas les TDL, car il est difficile de maintenir cette liste à jour, par exemple lorsque les équipements dans les salles tombent en panne ou sont remplacés.

Après avoir présenté notre système de création de DCP chiffrés, venons-en à notre deuxième contribution, qui concerne le déchiffrement.

5.3.2 Déchiffrement d'un DCP

Nous avons utilisé le module de gestion de DCP présenté dans le chapitre 3. Nous y avons ajouté un nouveau fichier source (dcpdecrypt.cpp).

Comme dans le chapitre 3, nous nous sommes tout d'abord confrontés à un problème de choix de bibliothèque et de compatibilité de licences. La première idée consistait à utiliser la bibliothèque OpenSSL pour la gestion du chiffrement RSA, car son usage est très répandu, mais intégrer OpenSSL dans le code de VLC provoque un problème de

compatibilité de licences (cf. chapitre 3). Nous avons donc choisi de tester deux autres bibliothèques compatibles avec la licence GPLv2 de VLC : Nettle et gcrypt.

L'utilisation de Nettle a été un échec, car cet outil ne gère pas le remplissage RSA-OAEP imposé par la SMPTE. Avec gcrypt, on peut correctement traiter l'algorithme RSA avec les contraintes du cinéma numérique. Cette bibliothèque est également utilisée par d'autres modules de VLC. Enfin, le noyau de VLC fournit une API, notamment au travers de la fonction `vlc_gcrypt_init`, qui permet d'initialiser la bibliothèque en mode *thread-safe*.

Nous traitons toute la partie du déchiffrement dans un fichier `dcpdecrypt.cpp`. À l'initialisation du module, la fonction `dcpinit` est appelée. Elle parcourt tous les fichiers XML de méta-données du DCP. Lors du parcours du fichier DCP, si une clé de type *KeyId* est rencontrée, cela signifie que le DCP est chiffré. À ce moment-là, le KDM associé à ce DCP est recherché. Les données du KDM sont validées puis, lors du parcours de la partie chiffrée en RSA, les clés AES de déchiffrement du DCP sont extraites. Ces clés sont stockées en mémoire pendant la lecture du DCP, puis effacées.

Le déchiffrement AES de chaque trame des fichiers MXF est effectué par la bibliothèque ASDCPLib. Nous savons (cf. chapitre 3) que cette bibliothèque utilise OpenSSL pour le déchiffrement, et nous envisageons donc de modifier ASDCPLib pour ne plus utiliser OpenSSL, mais par exemple gcrypt.

L'implémentation de la lecture du KDM et du déchiffrement RSA est très technique car elle utilise plusieurs concepts comme les fichiers PEM, X.509, PKCS#1, OAEP, ... Le détail de ces implémentations n'est pas exposé ici, mais est consultable dans le code de VLC (fichiers `modules/access/dcp/dcpdecrypt.cpp` et `modules/access/dcp/dcpdecrypt.cpp`).

5.4 Évaluation

Nous montrons ici qu'il est possible de créer des DCP chiffrés en logiciel libre. Nous évaluons d'abord le logiciel DCP-o-matic dans le mode chiffrement, puis nous vérifions que nous pouvons lire le DCP chiffré avec VLC.

5.4.1 Évaluation du chiffrement de DCP

Pour évaluer le fonctionnement du chiffrement de DCP dans DCP-o-matic, nous avons testé la génération d'un DCP et la génération d'un KDM pour un équipement existant. Nous avons choisi l'équipement numérique de la salle 1 du cinéma Utopia de Toulouse, qui est équipée de matériel Doremi et d'un projecteur Christie.

Nous avons utilisé un DCP non chiffré existant comme source pour les tests, en l'occurrence le film-annonce de *Moonrise Kingdom*. Nous avons converti ce DCP au format MP4 pour le donner en entrée à DCP-o-matic, qui a bien généré un DCP chiffré.

Pour la partie KDM, nous sommes allés chercher le certificat des équipements Doremi de la salle 1 du cinéma Utopia de Toulouse. Ce certificat est disponible et librement accessible sur le site arcene du CNC⁴. Le certificat téléchargé n'a pas été reconnu par le logiciel

4. <https://www.cnc-arcene.fr/>

DCP-o-matic, mais il nous a permis de connaître le numéro de série de l'équipement de la salle 1. Grâce à ce numéro de série, nous avons pu récupérer le certificat sur le site de Doremi⁵. Avec ce certificat, nous avons pu générer un KDM.

La différence entre les certificats fournis par arcene et par Doremi porte sur le formatage du contenu du certificat (au format PEM). Le certificat provenant de Doremi est écrit sur 27 lignes, avec une largeur maximale de 64 colonnes par ligne, alors que celui provenant d'arcene est écrit sur 2 lignes. Mais les contenus des certificats sont identiques. Le formatage d'arcene découle en fait d'un problème d'encodage du retour à la ligne.

Nous avons ensuite injecté le DCP et le KDM dans l'équipement de projection de la salle 1 du cinéma Utopia de Toulouse. Le DCP et le KDM ont été correctement reconnus, et nous avons pu projeter le DCP.

5.4.2 Évaluation de la lecture de DCP chiffré avec VLC

Sur la machine dédiée à la lecture de DCP, nous avons généré la clé privée et le certificat avec un des outils de *Digital Cinema Tools Distribution*.

En appliquant la même méthode que dans le paragraphe précédent, nous avons créé un DCP chiffré pour le lecteur VLC. Nous avons utilisé le certificat généré dans la machine équipée du lecteur VLC. Nous avons ensuite copié le DCP et le KDM sur cette machine.

Nous avons lancé VLC en lignes de commande, en spécifiant les chemins vers le DCP et vers le KDM. Un exemple de lignes de commande est donné ci-après :

```
1 export DCP_DIR=/chemin/vers/dcp
2 export KDM_FILE=/chemin/vers/kdm
3 ./vlc dcp://$DCP_DIR --kdm=$KDM_FILE
```

Le DCP chiffré du film-annonce de *Moonrise Kingdom* a pu être lu, sans que nous observions de ralentissement de traitement dû au déchiffrement symétrique AES sur les trames MXF.

Nous aurions également souhaité tester la lecture d'un DCP chiffré avec un KDM issu d'un distributeur connu, pour avoir une autre source que DCP-o-matic. Mais les distributeurs contactés ne savaient pas injecter un certificat dans leur outil de gestion de KDM. L'outil est vu comme une boîte noire, et la gestion des certificats est faite par la société exploitant l'outil. Notre solution n'étant pas officielle, nous n'avons pas pu intégrer notre certificat dans l'outil.

Nous n'avons pas non plus testé la lecture d'un DCP sur un long métrage, c'est-à-dire sur un contenu comportant des changements de bobine virtuelle, donc potentiellement des changements de clé AES à chaque nouvelle bobine. DCP-o-matic ne permet pas la réalisation de DCP sur plusieurs bobines. L'outil OpenDCP comporte cette possibilité, mais ne prend pas en compte le chiffrement de DCP.

5. <http://support.doremitechno.org/index.php/downloads/certificates>

5.5 Bilan

Le DCI n'impose pas le chiffrement d'un DCP. Pourquoi donc les distributeurs de films Art et Essai passent-ils par le chiffrement de DCP ? Il se trouve qu'ils ont souvent plus besoin de garantir une large diffusion de leurs films en salle plutôt que de protéger les contenus contre le piratage. On trouve très peu de films avec le label « Recherche et Découverte » dans les sites de partage de fichiers illégaux. En fait, les distributeurs ont trouvé dans le chiffrement un moyen pratique pour s'assurer qu'un film est toujours diffusé dans les salles, en limitant la durée de validité des KDM à une ou deux semaines par exemple.

Nous avons réussi à montrer qu'il était possible de concevoir une chaîne logicielle libre pour le chiffrement et le déchiffrement de DCP. Nous avons montré qu'il existait des outils libres pour le chiffrement de DCP. D'autre part, nous avons implémenté des solutions de déchiffrement dans le module DCP de VLC. Il reste des fonctionnalités à développer ou à adapter (notamment pour la création de DCP chiffrés), afin de parvenir à une chaîne complète de traitement d'un DCP chiffré.

Notre but était de montrer la faisabilité de la lecture d'un DCP chiffré. Nous n'avons donc pas implémenté tous les types de sécurité exigés par le DCI. Cela constitue une perspective intéressante mais pas forcément aisée. En particulier, implémenter des traces sécurisées n'implique pas simplement d'appliquer les normes correspondantes de la SMPTE.

Le marquage de contenu constitue un sujet de recherche toujours d'actualité. Nous n'avons aucune certitude sur son utilisation dans les équipements de projection. Les fichiers KDM peuvent en théorie contenir des informations relatives au marquage de films. Dans tous les KDM analysés, nous n'avons jamais vu de telles informations, mais cela ne signifie pas non plus qu'il n'y ait pas eu de marquage.

Les deux derniers types de sécurité sont encore plus complexes à implémenter. Le *Security Manager* (SM) peut être vu comme un processus tournant sur la machine VLC et scrutant toutes les entrées/sorties, vu que la machine utilisée est un PC classique : le nombre de choses à surveiller peut rapidement croître (réseau, USB, ...). Notons qu'une des fonctions du SM est de traiter la validité de la TDL fournie par le KDM, or nous avons vu que cette liste était souvent vide dans le KDM (même pour le film *Vice Versa* distribué par Disney!).

Enfin, les exigences liées à la « sécurité physique » semblent difficiles à respecter pour la solution de projection proposée. Pour respecter ces exigences, il faudrait enfermer la machine de lecture dans un coffre, dont la clé serait accessible aux seules personnes autorisées, et désactiver la fonction VLC de conversion entre formats vidéo. Peut-être est-il plus raisonnable d'attendre un assouplissement de ces exigences de la part du CNC.

Signalons pour finir que la question du chiffrement des DCP n'est pas sans conséquence sur l'archivage des œuvres. En effet, le chiffrement présente le risque qu'à long terme, une œuvre ne puisse plus être lue, si l'on ne sait plus générer le KDM permettant de la déchiffrer.

Chapitre 6

Un réseau opérationnel de distribution de DCP

6.1 Introduction

Une fois réalisé par un producteur, un film est distribué. Nous étudions le cas de la distribution vers les cinémas (et non celui de la distribution par DVD ou VOD).

On peut envoyer un DCP physiquement, en le copiant sur un disque dur puis en l'expédiant par transporteur. Le DCP peut également être transporté sous forme dématérialisée, c'est-à-dire par les réseaux.

L'envoi de DCP physiques est encore très répandu à l'heure actuelle. La méthode de transport est directement héritée de la circulation des copies 35 *mm*. Des sociétés sont spécialisées dans cette activité. Elle gèrent des stocks de disques durs qui sont expédiés vers les salles de cinéma. Les salles de cinéma récupèrent le DCP sur le disque, puis font circuler le disque dur vers d'autres salles, selon un plan de distribution défini par le transporteur.

Pour l'envoi dématérialisé de DCP, il existe principalement deux méthodes : le satellite et le réseau Internet. Le transport par satellite est une solution de type *broadcast* : on charge le DCP une seule fois, puis plusieurs lieux peuvent le récupérer simultanément. Au regard des coûts de distribution par satellite, ce sont surtout les grands distributeurs et les multiplexes qui utilisent cette méthode. Elle n'est pas économiquement viable pour le cinéma d'Art et Essai.

L'autre solution de transfert dématérialisé de DCP consiste à utiliser le réseau Internet. Cette solution est moins coûteuse, mais plus complexe à mettre en œuvre car elle est tributaire de la bande passante réseau de chaque salle. Nous utilisons ce type de transport pour notre solution de transfert de DCP, en utilisant des techniques « pair à pair » (*peer-to-peer*) de transfert de fichiers.

6.2 Deux notions clés pour la distribution de DCP

6.2.1 Fonctionnement d'un CDN

Une architecture réseau dédiée à la distribution de contenu spécifique est généralement nommée *Content Delivery Network* (CDN). Le concept de CDN est naturellement né avec la croissance du web et l'explosion du nombre de contenus à haut débit (vidéo par exemple) servis par Internet. Un clip vidéo très populaire peut aujourd'hui être lu simultanément depuis des milliers de villes répartis sur le globe. Répondre à ces requêtes depuis un serveur unique est inadapté (des problèmes de bande passante et de latence se posent). Un CDN est une infrastructure plus complexe, qui inclut de nombreux serveurs dont la répartition doit être cohérente avec celle des requêtes. Les contenus à transmettre sont ainsi répartis de manière à ce qu'ils soient délivrés aux clients, c'est-à-dire aux utilisateurs, qui souhaitent y accéder. L'infrastructure est cachée du point de vue du client : un utilisateur peut penser accéder à un serveur unique, éventuellement très éloigné, au travers d'URL mais, en réalité, des dispositifs matériels et logiciels servent d'intermédiaires. Le but de ces derniers est de réaliser un compromis entre la qualité de service (diminuer les latences d'accès ou la congestion des réseaux) et leur coût (répliquer des contenus et multiplier les utilisateurs coûte cher).

Les composants d'un CDN sont [23] :

- Des « serveurs de substitution » (*Surrogate Server* (SS)) dupliquant les contenus, issus du « serveur d'origine » (*Origin Server* (OS)).
- Une infrastructure réseau entre les sites du CDN.
- Un système de routage pour distribuer les requêtes du client final, sur la base de plusieurs facteurs (comme la présence du contenu, la charge des serveurs, la congestion du réseau). Les équipements de routage sont également appelés les « décideurs » (*Decision Maker* (DM)) : leur but est de router les requêtes vers le meilleur site, en fonction d'une métrique de performance donnée.
- Une infrastructure de livraison pour distribuer le contenu des SS vers le client.
- Des règles pour distribuer et gérer le contenu entre l'OS et les SS.
- Un système de gestion pour collecter des données statistiques et la facturation.

Un CDN est donc un réseau superposé (*overlay network*). La fonction des éléments de ce réseau est de stocker, distribuer et livrer (« router ») du contenu.

Au-delà de la multiplication et de la répartition des serveurs dont nous venons de parler, le modèle pair à pair propose un pas supplémentaire en matière de décentralisation d'un service. L'idée initiale est que chaque client puisse aussi devenir un fournisseur du contenu qu'il demande. On dira qu'un pair est à la fois client et serveur. Cette idée de mutualisation, de partage des ressources et des contenus est à l'origine des applications pair à pair à large échelle permettant le partage de contenu, la diffusion de flux multi-points, le stockage persistant, ... Il faut comprendre que le niveau de décentralisation (par exemple la complexité et la dynamique des réseaux logiques interconnectant les pairs) mis en œuvre dans une architecture pair à pair dépend de l'application cible.

Pour notre problème de distribution de DCP, nous avons choisi d'utiliser une application pair à pair, en l'occurrence le protocole BitTorrent. Cette application, utilisée depuis

longtemps, supporte le transport de fichiers de grande taille. Elle est distribuée et multi-plateforme. Elle possède également un système de découpage en morceaux (*chunks*) et un mécanisme de contrôle d'intégrité qui correspondent à nos besoins.

6.2.2 Protocole BitTorrent

BitTorrent est une application pair à pair dont le but est de faciliter le transfert de fichiers. L'idée de base pour le transfert BitTorrent est de diviser un fichier volumineux en petit morceaux. Un fichier de 100 Mo est généralement divisé en morceaux de 256 ko. Les pairs n'ayant pas encore reçu le fichier cherchent à télécharger simultanément les morceaux de fichier manquants. L'algorithme pour la distribution des morceaux entre pairs est fondé sur le principe du « tac au tac » (*tit for tat*) [14].

Une somme de contrôle est calculée pour chaque morceau de fichier. L'identification des morceaux et leur somme de contrôle associée sont stockés dans un fichier Torrent. Ce fichier généré est distribué aux pairs souhaitant télécharger le contenu. On ajoute au fichier Torrent le nom du fichier et le traqueur lié au téléchargement du fichier.

Pour faciliter le processus, BitTorrent utilise un logiciel centralisé appelé « traqueur » (*tracker*). Dans un réseau BitTorrent, également appelé « essaim » (*swarm*), un pair voulant télécharger un fichier se connecte au traqueur associé au fichier. Le traqueur retourne une liste aléatoire de pairs possédant le fichier (partiellement ou complètement). L'application qui télécharge établit une connexion avec chacun des pairs de la liste. et recherche les morceaux de fichier disponibles. L'application fait la requête des morceaux de fichier qu'elle ne possède pas encore à tous les pairs avec lesquels elle s'est connectée. Chaque pair est autorisé à télécharger un nombre limité de fichiers par unité de temps. L'envoi d'un morceau de fichier est appelé *unchoking* dans le protocole BitTorrent. La sélection des pairs autorisés à recevoir le morceau de fichier est effectuée en fonction de leurs débits de téléchargement. Plus ce débit est élevé, plus le pair est prioritaire. Cependant, pour que l'on n'envoie pas de morceaux de fichier uniquement à ceux qui ont le débit le plus élevé, il existe un mécanisme dit *optimistic unchoking*, grâce auquel le pair qui envoie des morceaux sélectionne aléatoirement des pairs de destination pour, là encore, une période de temps donnée.

BitTorrent distingue deux types de pairs : ceux qui téléchargent un fichier, appelé *leechers* (« sangsues »), et ceux qui disposent du fichier complet pour la distribution, appelé *seeders* (« semences »).

Dans les usages « classiques » de transfert de fichiers par BitTorrent, il y a plusieurs centaines de *seeders*, et plusieurs dizaines, voire plusieurs centaines de milliers de *leechers*.

Pour transférer des DCP vers les salles de cinéma, nous utilisons le protocole BitTorrent dans un réseau plus contraint que celui d'Internet (usage « classique » du protocole). Des machines sont identifiées à l'avance comme étant des *seeders*. Elles se trouvent dans des *data centers* et sont dotées d'un débit réseau élevé. Les *leechers* sont les salles de cinéma. Ils peuvent coopérer dans la distribution de DCP, mais dans la grande majorité des cas, leur débit en téléchargement est négligeable par rapport à celui des *seeders* en *data center*. Autre spécificité, nos *seeders* ne disparaissent jamais de l'essaim, comme cela est souvent le cas avec le transfert de fichiers classique. Nous utilisons le protocole

BitTorrent sans le modifier, mais il est exécuté dans un environnement plus contrôlé que les systèmes classiques de partage de fichiers.

Notre proposition s'inscrit dans une démarche similaire à celle proposée par des chercheurs italiens qui se sont également penchés sur le problème de la transmission des DCP aux exploitants. Nous décrivons leurs travaux dans le prochain paragraphe.

6.3 Présentation du CDN ED-CINE

Un CDN pour la distribution de DCP est présenté dans [18]. Nous détaillons et analysons ici le fonctionnement de ce CDN, dénommé ED-CINE.

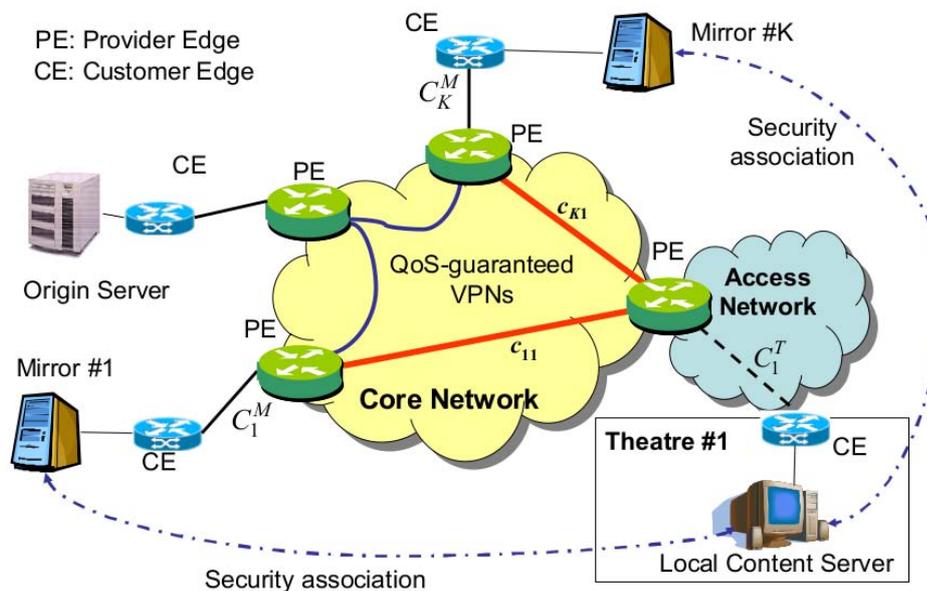


FIGURE 6.1 – Distribution de contenu avec le CDN ED-CINE (source [18]).

Le CDN ED-CINE est organisé selon une hiérarchie à trois niveaux (cf. figure 6.1), composée de : l'OS, où le contenu d'origine est stocké; les miroirs (les SS, dans la dénomination CDN), où ce contenu est dupliqué; les serveurs dans les cinémas, qui stockent les films demandés par les exploitants.

Les liens connectant l'OS vers les miroirs, et les miroirs vers les cinémas, sont des liens logiques, consistant chacun en un chemin composé de plusieurs liens physiques du réseau sous-jacent supportant les services du CDN. Ces liens logiques, qui peuvent être vus comme faisant partie d'un réseau privé (le CDN *overlay*), sont implémentés à travers des connexions VPN (*Virtual Private Network*).

Dans le système ED-CINE, le chiffrement est une exigence pour le transport de contenu du cinéma numérique. En outre, le transfert de DCP devant être sûr, il est

nécessaire d'effectuer le transfert via des applications utilisant le protocole fiable *Transmission Control Protocol* (TCP). Cependant, TCP adapte généralement le débit de réception à la congestion du réseau, ce qui implique des débits variables et un temps de téléchargement non prévisible. Pour pouvoir certifier une durée maximale de téléchargement, il est donc nécessaire de fournir des liens VPN avec un débit garanti.

Un nœud *Provider Edge* (PE) est un composant qui connecte le client au réseau du fournisseur d'accès. Le cinéma est dans un réseau privé local (i.e. LAN), la passerelle qui connecte le *Customer Edge* (CE) au PE est une connexion à haut débit (VDSL2, fibre optique, satellite, ...).

6.3.1 Procédure de routage des requêtes

Pour limiter le coût de gestion du réseau, les auteurs de [18] supposent que les liens VPN entre miroirs et cinémas sont configurés à la demande, avec une allocation de bande passante dépendant du service demandé.

Une fonction de base du CDN est de servir les requêtes de routage, le but étant de router les requêtes de contenu vers les miroirs les plus appropriés. Les auteurs de [18] supposent qu'il existe un DM central capable de collecter les requêtes des exploitants (e.g., via des interfaces web) et de connaître l'état du réseau sous-jacent (i.e., la disponibilité de la bande passante). Le DM est chargé d'exécuter les algorithmes de décision pour distribuer les requêtes de films vers les miroirs.

Il faut noter que le *mirroring* ne fournit pas seulement plusieurs sites différents, depuis lesquels on peut télécharger du contenu (ce qui augmente leur disponibilité, et contrôle les congestions des serveurs et du réseau), mais cela permet également de télécharger simultanément des morceaux de grands fichiers depuis un ensemble de miroirs, ce qui diminue le temps de téléchargement d'un contenu [36].

En fait, les clients perçoivent un débit de réception égal à la somme des débits des serveurs, et exploitent ainsi l'intégralité de la bande passante de leur connexion réseau.

En outre, le téléchargement parallèle garantit de la robustesse vis-à-vis des pannes de serveurs, et s'avère préférable dans les cas suivants : fichier de grande taille ; contenu disponible simultanément sur plusieurs serveurs ; exigences fortes en termes de durée maximale de téléchargement ; client équipé d'une connexion réseau à haut débit.

Le transfert de DCP possède les caractéristiques permettant le téléchargement parallèle. Les films peuvent être divisés en morceaux (*chunks*). Cependant, le coût de gestion global pour du téléchargement en parallèle peut être élevé. Il est nécessaire de configurer et de gérer plusieurs connexions VPN (les connexions VPN étant des connexions point à point, il faut initier plusieurs connexions VPN entre les miroirs et le client de réception) et de traiter les données pour fusionner les divers morceaux de films. L'usage d'un tel mécanisme doit être évalué avec attention.

Le processus de routage des requêtes pour le VPN ED-CINE est présenté ci-dessous :

- Le projectionniste/administrateur du cinéma demande, par exemple à travers une interface web, un ensemble de films, en tenant compte de l'espace de stockage disponible dans les serveurs du cinéma et du calendrier des projections. Cette

opération est autorisée seulement durant des intervalles de temps de réservation dans la journée. Le processus de réservation permet de choisir la durée maximale de téléchargement, parmi un ensemble de valeurs prédéfinies. Les auteurs de [18] supposent que cette durée est divisée en intervalles de temps (*time slots*). Le téléchargement peut occuper un ou plusieurs intervalles de temps. Les requêtes qui arrivent durant une période de réservation seront traitées au début du prochain intervalle de temps.

- Les requêtes venant des cinémas sont traitées par le DM, qui est également chargé de trouver les disponibilités des ressources en réseau.
- Dès que l'état du réseau est connu et que les demandes de distribution ont été traitées, le DM doit exécuter l'algorithme de routage des requêtes de transfert, pour gérer les requêtes arrivées durant la dernière période de réservation. Une solution consiste à allouer des ressources réseau et des miroirs avec les morceaux de films pertinents (cf. prochain paragraphe) pour chaque requête.
- Les connexions VPN et les configurations des serveurs sont définies selon la solution optimale.
- Les téléchargements démarrent au début du prochain intervalle de temps.

6.3.2 Modèle mathématique du CDN ED-CINE

Ce paragraphe présente le modèle mathématique proposé par les auteurs de [18] pour optimiser le coût de la distribution de films (DCP). Le but est de minimiser le coût opérationnel du réseau, tout en respectant les contraintes de bande passante et les durées de téléchargement. La table 6.1 résume les entrées/sorties du modèle.

Les contraintes du problème sont listées ci-dessous :

$$\sum_{i=1}^N c_{ji} \leq C_j^M, \quad j = 1, \dots, K \quad (6.1)$$

$$\sum_{j=1}^K c_{ji} \leq C_i^T, \quad i = 1, \dots, N \quad (6.2)$$

$$\sum_{j=1}^K D_{jil} = D_l u_{il} \quad \forall (i, l) \quad (6.3)$$

$$D_{jil} \leq \tau_{il} \cdot c_{jil}, \quad \forall (j, i, l) \quad (6.4)$$

$$c_{ji} \leq \bar{c}_{ji}, \quad \forall (j, i) \quad (6.5)$$

$$c_{ji} = \sum_{l=1}^H c_{jil} \quad (6.6)$$

$$0 \leq c_{jil} \leq v_{jl} \cdot u_{il} \cdot \bar{c}_{ji}, \quad \forall (j, i, l) \quad (6.7)$$

$$0 \leq D_{jil} \leq v_{jl} \cdot u_{il} \cdot D_l, \quad \forall (j, i, l) \quad (6.8)$$

TABLE 6.1 – Entrées/sorties du modèle.

Paramètres	Description	
Entrées	K	Nombre de miroirs (indexés par j)
	N	Nombre de cinémas (indexés par i)
	H	Nombre de films (indexés par l)
	D_l	Taille du film l
	C_i^T	Vitesse du lien d'accès (débit entrant) du cinéma i
	C_j^M	Vitesse du lien d'accès (débit sortant) du miroir j
	\bar{c}_{ji}	Bande passante maximale qui peut être allouée par le VPN du miroir j au cinéma i
	τ_{il}	Limite du temps de téléchargement pour le film l demandé par le cinéma i ($\tau_{il} \geq D_l/C_i^T, \forall(i, l)$)
	u_{il}	Variable binaire égale à 1 si le cinéma i a demandé le film l , à 0 sinon
	v_{jl}	Variable binaire égale à 1 si le miroir j stocke le film l , à 0 sinon
Sorties	c_{jil}	Capacité allouée à la connexion VPN entre le miroir j et le cinéma i pour télécharger le film l (éventuellement un morceau)
	D_{jil}	Taille du fragment du film l récupéré par le cinéma i depuis le miroir j
	c_{ji}	Quantité totale de capacité allouée sur le VPN depuis le miroir j vers le cinéma i ($c_{ji} = \sum_{l=1}^H c_{jil}$)

Les équations (6.1) et (6.2) traduisent les contraintes de bande passante sur le miroir et sur le cinéma, respectivement. La contrainte d'intégrité sur chaque requête de film se traduit par l'équation (6.3), tandis que la contrainte sur la durée maximale de téléchargement est traduite par (6.4). Les contraintes sur la bande passante et sur la quantité de bande passante allouée sur chaque VPN($j \rightarrow i$) sont traduites par les équations (6.5) et (6.6), respectivement. Enfin, la plage de la quantité de bande passante allouée sur le VPN($j \rightarrow i$) pour le film l , et la plage de la taille des *chunks* appropriés, sont définies par (6.7) et (6.8), respectivement. La fonction de coût à minimiser s'écrit :

$$F_{\text{Total}} = \sum_{(j,i)} f_{ji}(c_{ji}) \quad (6.9)$$

où $f_{ji}(c_{ji})$ est le coût des connexions VPN(j, i). La fonction f_{ji} doit être croissante et vérifier $f_{ji}(0) = 0$. Comme, de plus, elle doit inclure le coût initial d'installation du VPN,

elle peut par exemple être définie comme suit :

$$f_{ji}(c_{ji}) = \begin{cases} a_{ji} + g_{ji}(c_{ji}) & \text{si } c_{ji} > 0 \\ 0 & \text{si } c_{ji} = 0 \end{cases} \quad (6.10)$$

où $a_{ji} > 0$ et g_{ji} est une fonction croissante telle que $g_{ij}(0) = 0$. Il faut noter que f_{ji} est non convexe et non continue en 0. Le choix d'une telle fonction de coût garantit que le téléchargement en parallèle n'est activé que si cela est strictement nécessaire. Si les fonctions de coût sont homogènes (i.e., $f_{ji}(c_{ji}) = f(c_{ji}), \forall (j, i)$), alors la minimisation du coût global est équivalente à la minimisation du nombre de connexions VPN à initier. Il est également notable que la fonction de coût (6.9) est une combinaison linéaire de fonctions discontinues non convexes, et donc le problème n'est ni continu, ni convexe. Même des outils comme LINGO¹ et AMPL/MINOS² ne peuvent pas garantir la convergence vers un optimum global.

6.4 Conception d'un nouveau CDN

Le travail que nous venons de décrire brièvement est plutôt théorique. Il permet de dimensionner les composants du CDN, mais ne propose pas de solution pratique pour la mise en œuvre d'un CDN de DCP.

6.4.1 Description des besoins

Le système de distribution que nous souhaitons mettre en œuvre doit pouvoir distribuer 1000 DCP vers 200 établissements. La taille moyenne d'un DCP étant 200 Go, il faut être capable de transporter 200 To de contenu en un mois.

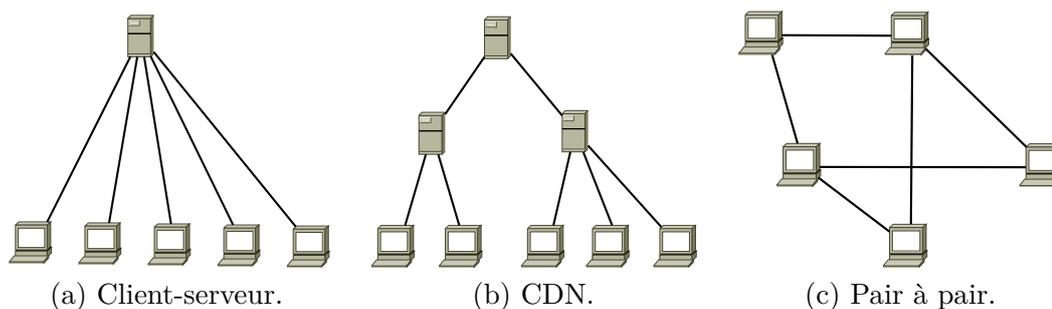


FIGURE 6.2 – Systèmes de distribution réseau de contenus.

1. LINGO : *Optimization Modeling Software for Linear, Nonlinear, and Integer Programming* (http://www.lindo.com/index.php?option=com_content&view=article&id=2&Itemid=10).

2. MINOS est un solveur pour le langage de modélisation AMPL (<http://ampl.com/products/solvers/solvers-we-sell/minos/>).

La distribution avec un serveur unique (cf. figure 6.2-a) est totalement inadaptée, vu le volume de données à transporter. Même si la mise en œuvre d'un CDN (cf. figure 6.2-b) est adaptée au volume de données à transporter, elle nécessite de déployer trop de serveurs secondaires. Le modèle pair à pair (cf. figure 6.2-c) est théoriquement adapté, mais peu réaliste. Les DCP sont transportés vers des cinémas, qui n'ont pas une bande passante réseau suffisamment élevée pour assurer le transport des DCP. La majorité des cinémas possèdent des connexions de type *Asymmetric Digital Subscriber Line* (ADSL). Une connexion ADSL garantit un débit de réception de l'ordre de 10 à 20 Mb/s, mais le débit d'émission ne dépasse généralement pas 5 Mb/s. Par conséquent, une connexion ADSL permet de recevoir une grande quantité de données, mais il est plus difficile d'envoyer des données car la bande passante d'émission est limitée.

Dans une application pair à pair classique de partage de contenu, comme The Pirate Bay, l'échange de contenu fonctionne bien, même avec des connexions xDSL, car le réseau n'a pas la même échelle. Dans The Pirate Bay, il y a plusieurs milliers de contenus, de tailles faibles par rapport à celle d'un DCP (de 700 Mo à 4 Go), qui sont partagés entre plusieurs dizaines de milliers d'utilisateurs.

L'architecture du réseau de distribution que nous proposons est hybride entre CDN et pair à pair (cf. figure 6.3). Comme dans un CDN classique, l'OS permet d'ajouter un contenu dans le réseau logique. Les serveurs secondaires permettent de dupliquer le contenu. La duplication des contenus est faite pair à pair. Les cinémas peuvent récupérer du contenu sur n'importe lequel des serveurs du réseau logique.

Sur la figure 6.3, nous avons représenté les liens entre un cinéma et deux autres cinémas par des traits en pointillés. Ce cinéma est équipé d'une liaison réseau à très haut débit (par exemple, une connexion en émission et réception à 100 Mb/s). Dans ce cas, il peut activement contribuer au partage de ressources des DCP qu'il est en train de récupérer. Notons ici une autre différence avec le pair à pair pour le partage de contenu décentralisé : dans notre cas, seuls les serveurs ont vocation à stocker les DCP.

6.4.2 Comparaison avec l'architecture du CDN ED-CINE

The DCP Bay a le même objectif que ED-CINE : distribuer des DCP vers des salles de cinéma en respectant les contraintes de bande passante et les durées maximales de téléchargement. Alors que ED-CINE propose un modèle de distribution, nous disposons d'un système opérationnel. ED-CINE ne donne pas de résultat de distribution réelle, mais propose un modèle de distribution cherchant à optimiser le coût de distribution des DCP.

Au-delà de cet objectif similaire, l'architecture de notre CDN et celle de ED-CINE présentent des similarités. D'une part, nous utilisons un OS et des SS (qui sont appelés « miroirs » dans ED-CINE). D'autre part, le transfert des DCP est sécurisé, dans les deux cas, par la mise en place de connexions VPN.

Le choix du VPN, dans les deux cas, est de type *overlay*. Notre CDN comporte néanmoins une différence fondamentale avec ED-CINE : alors que ce dernier sous-traite à un fournisseur réseau la mise en place des liaisons VPN, cela fait partie intégrante de notre CDN. Dans le cas de ED-CINE, le fournisseur réseau certifie une « qualité de

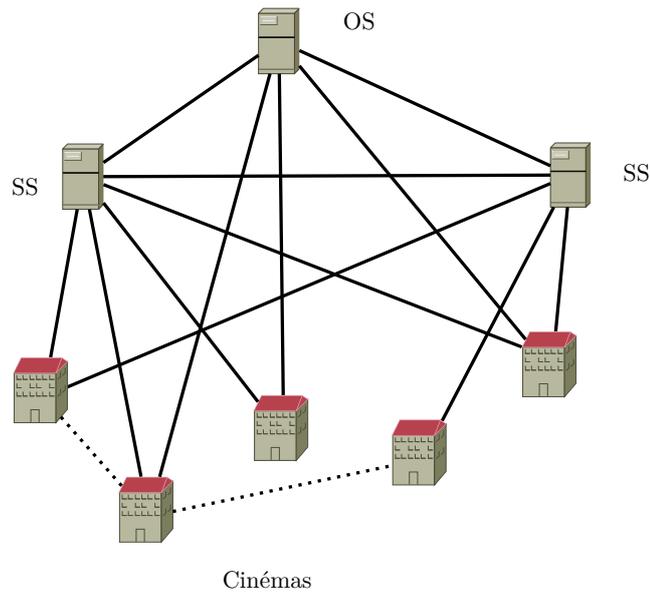


FIGURE 6.3 – Architecture hybride de The DCP Bay, entre CDN et pair à pair.

service », c'est-à-dire la mise à disposition d'une bande passante garantie. Dans le cas de The DCP Bay, il est plus difficile, vu l'architecture, de garantir une bande passante, mais le recours à des prestataires réseau fournissant une bande passante garantie est trop coûteux, au regard de nos besoins de distribution de DCP.

Avec le système ED-CINE, il peut y avoir une augmentation du coût opérationnel en cas de transfert en parallèle. Nous ne courons pas ce risque, car la distribution utilise le protocole BitTorrent, qui inclut par nature le transfert de données en parallèle. Le VPN Tinc étant intégré et distribué dans notre système, la création de liaisons VPN ne cause pas de coûts supplémentaires.

Le système ED-CINE fonctionne par intervalles de temps pour la distribution, afin de minimiser le nombre de connexions VPN simultanées. Vu que nous n'avons pas cette contrainte sur le VPN, nous n'avons donc pas besoin de définir des intervalles de temps. L'algorithme de décision du DM s'en trouve simplifié. Notons que pour l'instant, le DM de The DCP Bay est humain. La question qu'il se pose est la suivante : « Ai-je assez de bande passante pour transférer le DCP dans le temps imparti ? » Il évalue la bande passante sortante des serveurs et la bande passante en entrée du cinéma. Le logiciel Tuco (cf. paragraphe 6.5.1) retourne des informations qui aident le DM à prendre des décisions. Ce logiciel fournit notamment le nombre de demandes de DCP simultanées : ceci permet de vérifier que la bande passante sortante des serveurs est suffisante. De plus, Tuco fournit des statistiques sur les durées de téléchargement de DCP de chaque salle, ce qui permet de prévoir si la bande passante de réception du cinéma est suffisante.

Le modèle mathématique proposé par ED-CINE ne prend pas en compte, dans la mise à disposition d'un film par un distributeur, le moment où le contenu arrive effectivement

dans l'OS. Ceci peut être une contrainte forte pour respecter les durées de téléchargement et l'allocation de bande passante. Notons aussi, même si cela semble plutôt complexe à modéliser, que le modèle de ED-CINE ne prend pas en compte les téléchargements simultanés chez les exploitants depuis plusieurs services de transport dématérialisé. Il est supposé, dans ce modèle, que la connexion réseau est uniquement dédiée au CDN ED-CINE, comme l'impose par exemple la solution Globecast. Un cinéma ne s'abonne pas à une connexion réseau par transporteur de contenu, car cela lui coûterait trop cher. En pratique, un cinéma partage sa connexion réseau entre plusieurs transporteurs.

6.4.3 Fonctionnement du système de distribution

Le système de distribution que nous proposons est représenté sur la figure 6.4.

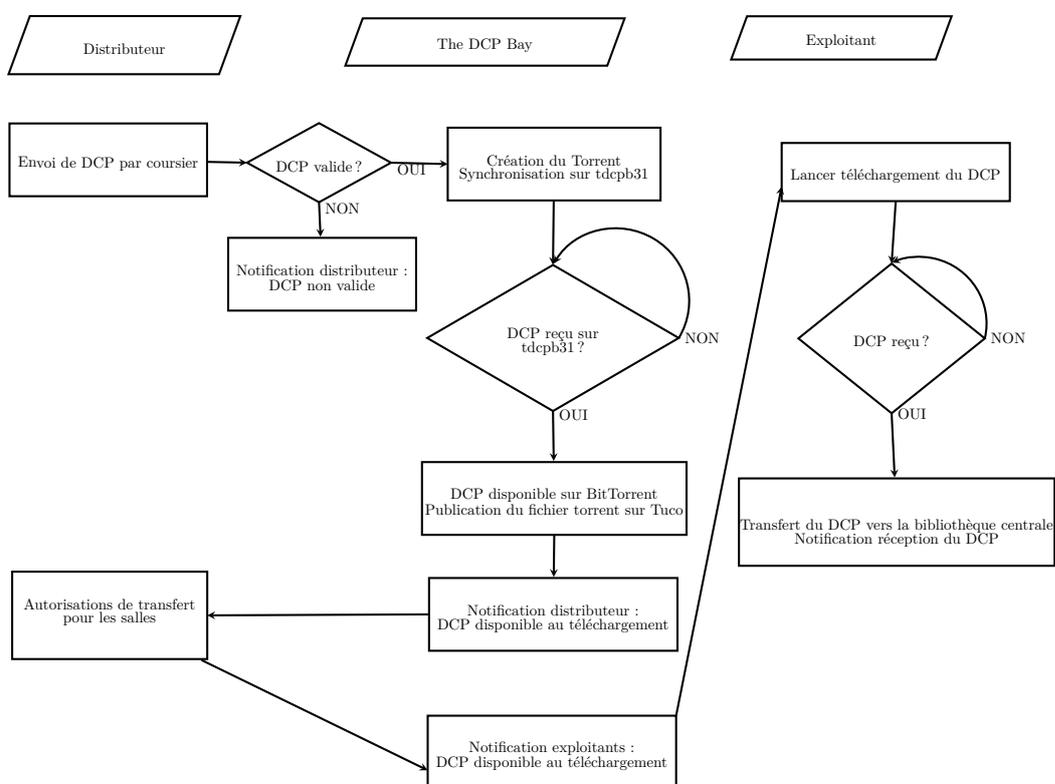


FIGURE 6.4 – Diagramme de séquence du système The DCP Bay.

Plusieurs acteurs interviennent dans le système de distribution : les distributeurs, les exploitants et le transporteur. Le transporteur est en fait la plateforme technique de The DCP Bay, qui supervise le transport de DCP par des actions soit automatiques, soit humaines. La supervision du système de distribution est effectuée par l'application Tuco (cf. paragraphe 6.5.1), qui permet aux distributeurs et aux exploitants de gérer le transfert de DCP.

Une fois le DCP livré par coursier au GNCR, le film est copié sur la machine tdcpb. Son intégrité est vérifiée, puis le fichier Torrent est créé, et le DCP est copié en *seed* (« graine ») sur la machine du GNCR. Le Torrent est ensuite dupliqué, au moins sur tdcpb31. Une fois le transfert effectué, le fichier Torrent généré est chargé dans l'application Tuco, et des méta-données sont collectées. Enfin, le distributeur est averti que le film est disponible au téléchargement.

Le distributeur se connecte sur l'application Tuco et sélectionne les salles à qui il veut autoriser le chargement du DCP. Une fois que le film est autorisé au chargement par le distributeur, l'exploitant reçoit un courriel le prévenant de la disponibilité du DCP, qui comporte le fichier Torrent associé au DCP en pièce jointe. L'exploitant peut alors lancer le chargement du DCP.

Il nous a été demandé de mettre en place ce mécanisme d'autorisation car l'exploitant souhaite connaître à l'avance ce qu'il va recevoir dans sa bibliothèque. Il arrive fréquemment, avec d'autres solutions de transport dématérialisé, qu'un film arrive dans la bibliothèque d'un cinéma sans que son exploitant l'ait demandé, ni même programmé. Les exploitants sont gênés par ce mécanisme, car la réception de films non demandés réduit la bande passante pour le transfert d'un film demandé, et occupe inutilement de l'espace disque dans la bibliothèque.

Ce mécanisme d'autorisation permet aussi au distributeur de contrôler vers quelles salles le DCP sera expédié. Chaque transfert de DCP a un coût, qui est à la charge du distributeur. L'exploitant prend à sa charge le coût des lignes d'accès à Internet et le coût d'adhésion annuelle à notre réseau de distribution. Les coûts sont ainsi répartis entre exploitants et distributeurs.

Le distributeur, à travers l'application Tuco, peut vérifier à quels moments le téléchargement du DCP a débuté et s'est achevé. L'exploitant peut lui aussi se connecter sur Tuco : le site lui fournit un récapitulatif de tous les téléchargements effectués. Il peut télécharger soit des DCP de Film-Annonce (FA), soit des DCP de films. L'exploitant ne peut télécharger de DCP de films que si le distributeur lui en a donné l'autorisation, mais il peut télécharger les FA sans l'autorisation du distributeur. L'exploitant a donc une vision complète du catalogue. Il peut faire la demande auprès du distributeur d'un film pour lequel il n'a pas encore d'autorisation de téléchargement, en envoyant un courriel (sous son identité) au distributeur du film voulu, pour lui indiquer qu'il souhaite télécharger le film via notre plateforme.

Sur la machine de réception de chaque cinéma est installé un client BitTorrent. Pour lancer le transfert d'un film, l'exploitant a deux possibilités. Soit il utilise le fichier Torrent envoyé par courriel et le met en chargement sur son client Torrent, soit il lance le téléchargement par l'application Tuco. Le téléchargement peut se lancer directement depuis l'interface, car la machine hébergeant l'application Tuco se trouve sur le même réseau VPN Tinc que les machines des exploitants, et le client BitTorrent (Transmission) dispose d'une interface de type *Remote Procedure Call* (RPC) permettant de le piloter à distance.

Le début et la fin d'un téléchargement sont connus grâce au traqueur Torrent. Toutes les heures, l'application Tuco effectue une requête sur la base de données du traqueur

pour récupérer des données. Les données récupérées sont : le type d'événement (début ou fin de téléchargement), la date de l'événement, l'IP (Tinc) de l'origine de l'événement, et la somme de contrôle (*hash*) du Torrent qui a été mis en téléchargement.

Le lien avec la salle se fait au travers de l'IP Tinc. Le lien avec le DCP se fait par le *hash*, qui est unique. Le fichier Torrent de chaque DCP a un *hash* qui l'identifie. La base de données de l'application Tuco stocke le *hash* de chaque fichier Torrent téléchargé.

6.5 Mise en œuvre du CDN The DCP Bay

Le système est composé de machines en *data center* et de machines de réception dans les cinémas. Nous travaillons avec des FAI associatifs, auxquels les serveurs de The DCP Bay sont connectés.

6.5.1 Interfaces logicielles de supervision

Nous présentons ici les interfaces pour les utilisateurs de The DCP Bay.

Application Tuco

Tuco est l'interface web pour les exploitants et les distributeurs. L'application possède également une interface pour l'administration. Tuco a été développé en Flask, qui est un *framework* Python pour le développement d'applications web (<http://flask.pocoo.org/>). Nous avons choisi Flask car il s'agit d'un logiciel libre, mais aussi car toute la partie concernant la création et la gestion de site se fait en Python, avec des modèles HTML pour le rendu.

De nombreuses fonctionnalités d'applications web sont intégrées à Flask, sous forme de modules, comme la gestion de la base de données, l'accès sécurisé au site, l'envoi automatique de courriels, ...

La partie administration permet :

- D'ajouter des distributeurs (avec nom et adresse électronique).
- D'ajouter des exploitants (avec nom, adresse électronique, adresse et IP Tinc).
- D'ajouter des films (avec titre, titre original et date de sortie).
- De télécharger des fichiers Torrent pour un film. Un film peut être composé de plusieurs DCP : film-annonce et long métrage, mais aussi version sous-titrée, *Open CAPtion* (OCAP), ...
- De suivre les autorisations, le début et la fin d'un téléchargement.

À l'avenir, l'application Tuco connaîtra des évolutions. Le fichier Torrent ne sera plus expédié en pièce jointe du courriel d'autorisation de téléchargement. Seul un lien permettant de le lancer par le RPC du BitTorrent sera donné.

Client BitTorrent

Nous avons choisi Transmission, car il s'agit d'un client BitTorrent libre, gratuit et très répandu. Le logiciel prévoit aussi une interface web pour la gestion des téléchargements,

ainsi qu'une interface par lignes de commande (via une interface RPC).

La figure 6.5 montre l'interface web du client Torrent Transmission du cinéma Atalante de Bayonne. On constate qu'il a déjà téléchargé le film-annonce de *La Niña De Fuego*, car le titre (qui commence par `LaNinaDeFuego_TLR`) apparaît en vert. Le téléchargement de *Les Mille et une Nuits* (volume 2) est en cours : il reste 1 jour et 6 heures de téléchargement. Le cinéma accède au client Torrent via l'adresse `http://64.cinema.tdcpb.org`. Ce site est en accès public, mais est protégé par login/mot de passe. Comme nous l'avons déjà dit, le protocole Torrent s'exécute sur le réseau Tinc, ce qui le rend non accessible depuis l'extérieur. Nous avons mis en place un accès public à l'aide d'une machine qui exécute un serveur web sur le réseau Internet, mais qui se trouve aussi sur le réseau Tinc. Ce serveur web a donc pour fonction de traduire et rediriger le flux de données d'une requête d'adresse web (ici : `64.cinema.tdcpb.org`) vers le port et l'adresse IP Tinc (ici : port 9091 sur l'adresse IP 10.10.10.64) du client Transmission de la machine du cinéma. Cette technique de redirection est appelée *reverse proxy*.

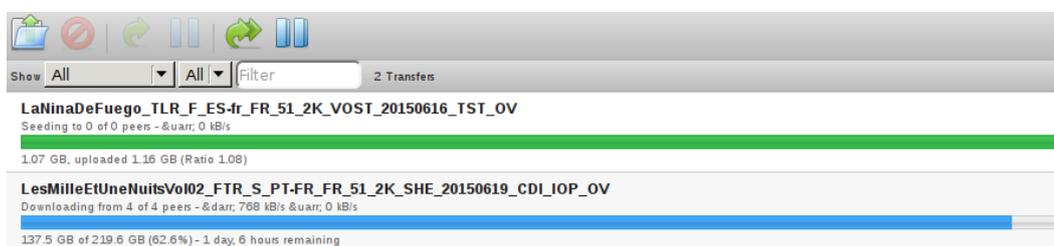


FIGURE 6.5 – Capture d'écran du client Transmission du cinéma Atalante de Bayonne.

Notre système fonctionne avec tout client BitTorrent, mais il faut néanmoins configurer son interface réseau sur le VPN Tinc.

À l'avenir, l'interface du client BitTorrent sera intégrée dans l'application Tuco. Cela facilitera l'utilisation de l'outil par l'exploitant, et nous permettra également d'évoluer, si nécessaire, vers d'autres clients Torrent.

6.5.2 Logiciels exploitant l'infrastructure matérielle

Les logiciels ayant permis le déploiement de l'infrastructure de The DCP Bay sont présentés ci-dessous.

VPN Tinc

Nous avons choisi le VPN Tinc, qui a comme particularités de travailler en pair à pair et de sécuriser le transfert, car il s'agit d'un VPN. Une comparaison de VPN libres, dont Tinc, est effectuée dans [24] : les avantages de Tinc sont qu'il est plutôt facile à mettre en place, et aussi que l'on peut assurer la maintenance d'un nombre élevé de machines sur le réseau.

On peut donc faire « tourner » un protocole pair à pair sur ce VPN. Nous avons configuré Tinc de manière à ce que toutes les machines du réseau se voient, comme si elles étaient sur le même réseau local, à travers un « tunnel réseau » pour chaque machine. Il y a un peu de perte de bande passante en passant par le VPN, mais on gagne en gestion du réseau. D'autres propriétés du VPN nous ont permis de réaliser l'agrégation de lignes réseau, que nous détaillerons un peu plus loin.

Traqueur

Il nous a également fallu mettre en place un traqueur BitTorrent pour gérer les demandes de contenu. Il existe de nombreux traqueurs libres, qui sont généralement utilisés par les plateformes de distribution de contenu comme The Pirate Bay. Le rôle du traqueur est simple : il répond à des requêtes sur le protocole HTTP. Il reçoit des requêtes lui demandant des listes de pairs possédant le contenu désiré, ou bien des indications sur le statut du pair (téléchargement fini, mise en pause, ...). Le traqueur stocke les données dans une base de données. Il utilise le *hash* du fichier Torrent comme clé pour identifier les différents fichiers à utiliser.

Nous avons choisi le traqueur XBT, qui est simple à mettre en place et utilise très peu de ressources sur la machine où il s'exécute.

Un traqueur écoute généralement les connexions sur Internet. Il peut être soit public, soit fermé par des méthodes d'authentification. Un traqueur public permet d'annoncer la liste de ses fichiers disponibles au téléchargement sans limitation. Un traqueur fermé limite la liste des utilisateurs, afin de contrôler qui est autorisé à télécharger un fichier.

Notre traqueur communique uniquement sur le réseau Tinc. Il ne répond pas aux demandes extérieures à ce réseau. Nous ajoutons manuellement des machines de réception dans le réseau. Nous n'avons alors pas besoin de traqueur fermé, car nous contrôlons les connexions des machines sur le réseau. De plus, la connexion sur le réseau Tinc est chiffrée, donc le transport de DCP l'est lui aussi.

6.5.3 Infrastructures

L'infrastructure de The DCP Bay est composée de FAI associatifs et de serveurs situés dans leurs *data centers*.

FFDN

Les FAI associatifs sont regroupés dans la FFDN, qui se définit ainsi :

« La FFDN regroupe des fournisseurs d'accès à Internet associatifs se reconnaissant dans des valeurs communes : bénévolat, solidarité, fonctionnement démocratique et à but non lucratif, défense et promotion de la neutralité du Net.

À ce titre, la FFDN se donne comme mission de porter la voix de ses membres dans les débats concernant la liberté d'expression et la neutralité du Net.

Elle fournit à ses membres les outils pour se développer et répondre aux problématiques qui concernent l'activité de fournisseur d'accès à Internet. »

TABLE 6.2 – Serveurs pour The DCP Bay.

Serveur	Localisation	Capacité de stockage (To)	Fonctions
gnrcr	GNCR	3	Serveur d'origine, <i>seeder</i>
tdcpb31	Tetaneutral	9	Serveur de substitution, traqueur
mirror	Tetaneutral	9	Serveur de substitution
utopialab	Tetaneutral	24	Archivage
tdcpbgitoyen	Gitoyen	3	Serveur de substitution
tdcpb1	Aquilenet	2	Serveur Tinc
tdcpbldn	LDN	1	Serveur de substitution
tdcpbgrenode	Grenode	1	Serveur de substitution

Nous travaillons avec la FFDN pour l'ensemble de ces raisons. L'entité gérant The DCP Bay est elle aussi associative et à but non lucratif.

Serveurs de distribution

Nous possédons plusieurs machines en *data center* (cf. table 6.2), qui sont connectées sur le même réseau VPN Tinc que chez les exploitants. Les principales caractéristiques des machines en *data center* sont leurs bandes passantes et leurs capacités de stockage.

Nous disposons de trois machines (tdcpb31, mirror et utopialab) hébergées à Tetaneutral. Elles forment, avec celle de Gitoyen, l'épine dorsale du système. La machine tdcpb31 stocke tous les nouveaux DCP, qui sont dupliqués sur mirror. La duplication sur mirror est virtuelle, dans le sens où mirror ne fait qu'un montage de partage de fichiers (NFS) sur tdcpb31. Ce mécanisme permet de facilement ajouter un *seeder* à l'essaim BitTorrent.

Les machines tdcpb31, mirror et tdcpb1 gèrent les connexions au VPN Tinc : elles stockent les clés publiques des autres machines qui se connectent sur le réseau.

La machine utopialab a une fonction d'archivage : le serveur stocke les DCP datant de plus de trois mois. Pour le stockage, on utilise le système de stockage distribué Ceph en mode *erasure coding*. Cela permet un stockage sûr des données. En effet, le système est robuste à la panne de disque dur et à la panne de machine. Le stockage des données est réparti sur plusieurs disques durs et sur plusieurs machines. De plus, l'ajout de nouvelles partitions se fait à la volée.

La machine située au GNCR à Paris permet de rentrer dans le système le DCP envoyé physiquement par coursier. Ces DCP seront ensuite dupliqués, au moins sur tdcpb31 et sur mirror. En effet, comme la majorité des distributeurs sont localisés à Paris, ils préfèrent souvent faire livrer leurs DCP par coursier. Nous proposons un service de transfert des DCP par FTP vers nos serveurs : ce service est finalement très peu utilisé, pour des raisons financières. Les distributeurs font réaliser leurs DCP par des laboratoires, et ces derniers facturent les envois par FTP, donc le coursier est plus économique.

Les machines tdcpbgitoyen, tdcpb1, tdcpbldn et tdcpbgrenode font office de caches, au cas où tdcpb31 et mirror seraient surchargées.

6.5.4 Équipement chez les exploitants

Pour pouvoir accéder au transfert dématérialisé de DCP, l'exploitant doit disposer d'un équipement possédant certaines caractéristiques. Il faut qu'il ait un accès à Internet, un espace de stockage suffisant et un accès au réseau du cinéma.

Caractéristiques du matériel

Nous proposons aux cinémas, soit de configurer du matériel déjà existant, soit de leur fournir une machine dédiée pour 350 € HT. Cette machine est un Intel NUC DN2820FYKH³. Elle est petite (base carrée de 10 cm de côté, pour une hauteur de 4 cm). Ce petit format présente un avantage, car les cabines de projection sont souvent exigües. On n'a besoin ni d'écran, ni de clavier pour contrôler le système. Cela se fait par des interfaces web (cf. paragraphe 6.5.1). De plus, vu que l'Intel NUC ne possède pas de ventilateur, il consomme très peu d'énergie. Une autre raison du choix de cette machine est son prix très abordable. Ces machines étant vendues sans RAM et sans disque dur, on doit ajouter :

- Une RAM de 4 Go, valeur standard de nos jours, largement suffisante.
- Un disque dur de 1 To, qui permet de stocker de quatre à cinq DCP. Le stockage est temporaire pour la réception du DCP, la destination finale étant soit la bibliothèque centrale, soit le serveur de projection.

Nous installons un système Debian/Linux, ainsi que les paquets logiciels pour le transfert du film (VPN, client BitTorrent).

Nous nous contraignons à proposer aux salles un système Linux (Debian ou Ubuntu), afin de nous en tenir à des solutions libres et aussi pour faciliter la maintenance. Nous fournissons aux exploitants les mots de passe d'administration de la machine, qui peuvent donc l'utiliser pour d'autres usages : site web, base de données, vidéo en poche, ...

Nous avons constaté que les cinémas faisaient appel à un installateur pour acquérir le matériel, assorti d'une maintenance annuelle. Les systèmes mis en place sont verrouillés, même pour les responsables des cinémas. Ils ne peuvent donc pas toujours utiliser leur matériel comme souhaité. La quasi-totalité des bibliothèques sont prévues pour GNU/Linux, mais l'installation de logiciels complémentaires est impossible. Le principal argument des installateurs est la sécurité du système. Le propriétaire du matériel ne peut donc pas l'utiliser comme il le souhaite.

Nous imposons à l'exploitant d'utiliser une machine dédiée au téléchargement. Cette machine est indépendante du système de projection déjà installée chez l'exploitant. Il est ainsi plus facile de s'interfacer avec tout système de projection.

Agrégation de lignes ADSL

Les salles de cinéma ont souvent des bandes passantes réseau trop limitées pour recevoir de nombreux DCP, notamment sur les liens réseau de type ADSL. Il est alors parfois nécessaire d'utiliser plusieurs lignes ADSL pour augmenter la bande passante. La méthode pour combiner plusieurs lignes s'appelle « l'agrégation de lignes ». Ce service est une option supplémentaire proposée par les FAI. L'agrégation se fait dans les équipements du FAI et avec un matériel dédié chez le client.

En raison du VPN et de l'architecture en pair à pair, nous proposons une solution d'agrégation de lignes réseau afin d'accroître la bande passante d'un cinéma. Elle a comme principal avantage d'être une solution purement logicielle, qui ne nécessite pas une configuration spéciale chez le FAI. Cette solution d'agrégation est représentée sur la figure 6.6.

3. <http://www.intel.fr/content/www/fr/fr/nuc/nuc-board-dn2820fykh.html>

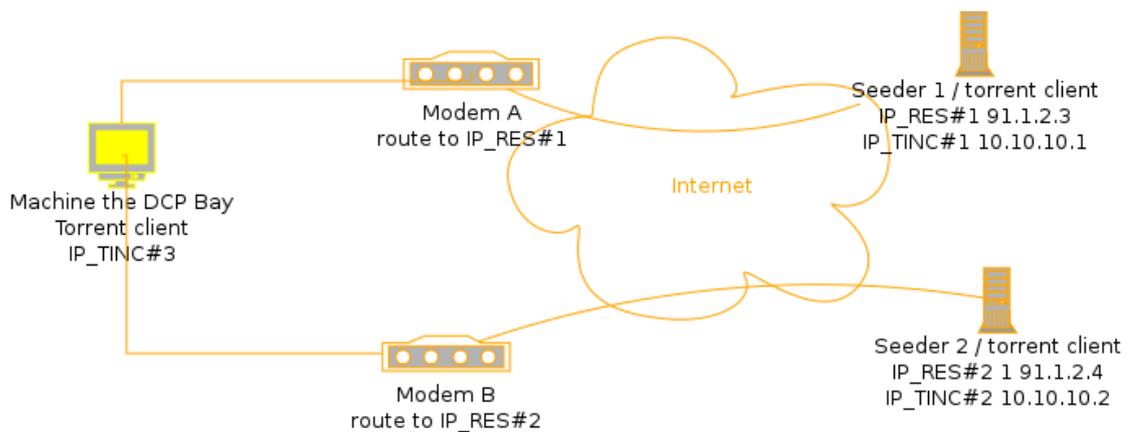


FIGURE 6.6 – Agrégation de lignes « The DCP Bay ». La machine « The DCP Bay » peut recevoir des données Torrent par les deux lignes réseau, en spécifiant les routes au niveau du VPN. Pour le client Torrent, toutes les machines se trouvent sur le même réseau VPN (10.10.10.XXX).

Pour chaque connexion réseau, il y a un modem. Dans la machine de réception de The DCP Bay chez l’exploitant, le contenu est reçu en pair à pair par au moins deux *seeders* dans nos *data centers*. Chaque *seeder* est un serveur possédant une adresse IP fixe. On crée alors des routes différentes sur la machine de réception : une vers le seeder #1 par le modem A, une autre vers le seeder #2 par le modem B. Ces routes sont définies au niveau IP standard, c’est-à-dire avec les tables de routage réseau de Linux. En revanche, le client Torrent a son interface sur le réseau du VPN Tinc. Le client va se connecter aux deux *seeders* pour recevoir des données en passant par les deux modems, donc par les deux lignes réseau du cinéma. Le réseau Tinc est « au-dessus » des tables de routage et ne voit pas que l’on ne passe pas par le même chemin. On utilise ainsi la bande passante des deux lignes. Ce système peut être étendu à plus de deux lignes, à condition qu’il y ait suffisamment de *seeders* disponibles en *data center*.

Cette solution d’agrégation de lignes est facile à mettre en place dans un cinéma, car la configuration des tables de routage se fait uniquement sur la machine de réception de DCP. Un autre avantage est que l’on peut avoir des FAI différents sur les deux lignes réseau, et que si un FAI est sujet à une panne, on peut toujours passer par l’autre ligne pour le transfert de contenu. Cette propriété ajoute de la robustesse, en comparaison des solutions classiques d’agrégation de lignes (i.e., celles proposées par les FAI ou bien celles de Globecast).

6.6 Évaluation

The DCP Bay est opérationnel depuis juin 2014. La table 6.3 indique le nombre de DCP qui ont été transférés depuis. Tous les cinémas connectés sont des salles d’Art et Essai équipées de un à sept écrans, mais certaines de ces salles ne se limitent pas à ce type de programmation et projettent des films grand public. Les types de connexion au réseau Internet dans les salles sont hétérogènes (fibre optique, VDSL, ADSL). Les usages eux aussi sont variables : certaines salles disposent de lignes Internet dédiées au transfert de DCP, alors que d’autres disposent d’une seule ligne Internet pour tous les usages, dont le transfert de DCP. Les exploitants peuvent être

TABLE 6.3 – Progression du transfert de DCP sur The DCP Bay.

	Nombre de films transférés	Nombre de films-annonces transférés	Données transférées (To)
Octobre 2014	48	0	6,4
Novembre 2014	39	0	4,7
Décembre 2014	41	0	5,9
Janvier 2015	50	0	5,8
Février 2015	73	5	8,7
Mars 2015	46	11	6,7
Avril 2015	28	8	2,8
Mai 2015	25	18	2,7
Juin 2015	42	25	5,7
Juillet 2015	25	16	3,2
Août 2015	31	41	4,0
Septembre 2015	77	38	9,9

équipés uniquement avec The DCP Bay, mais nous partageons souvent la connexion avec d'autres transporteurs de contenu. Les bandes passantes des salles sont donc très hétérogènes et varient au cours du temps, à cause du partage des connexions réseau. Suivant les salles, les scénarios de distribution de DCP ne sont pas tous les mêmes.

6.6.1 Évaluation du VPN Tinc

Tous les DCP transitent par des VPN et des tunnels (à l'exception du transfert par FTP). Le transfert de données par les tunnels n'atteint pas les mêmes performances, en terme de bande passante, que le transfert réseau direct. Nous avons directement connecté deux machines avec un câble Ethernet. Les deux machines ont des interfaces réseau Gigabit et s'exécutent sur la même version d'un système GNU/Linux. Nous avons mesuré le transfert de fichiers de 1 Go avec la commande `wget`. Avec la connexion directe, nous avons mesuré des débits de 960 Mb/s en moyenne. Via le VPN et le tunnel, nous avons mesuré des débits un peu moins stables, variant entre 220 et 250 Mb/s. Nous avons désactivé le chiffrement du VPN pour mesurer uniquement les performances du tunnel.

À un niveau plus macroscopique, c'est-à-dire dans des cas réels de transfert depuis une machine distante sur le réseau Internet, les débits de réception dans le tunnel ne sont pas toujours constants. Ce problème est probablement dû aux règles des pare-feux sur les modems réseau [39]. Ajoutons que nous ne travaillons pas avec des FAI dédiés dans les salles, que les modems fournis diffèrent beaucoup d'une salle à l'autre et atteignent des niveaux de performance variables.

Malgré la réduction de performance des tunnels, nous n'avons pas, à ce jour, été bloqués dans des transferts de DCP par la congestion des tunnels. Si on se donne une limite haute de 200 Mb/s pour une salle, cette salle peut recevoir un DCP de 200 Go en un peu moins de 3 heures, ce qui constitue une limite totalement compatible avec une rotation hebdomadaire des films.

De plus, l'utilisation du VPN Tinc facilite grandement la maintenance : toutes les machines sont vues comme si elles se trouvaient sur le même réseau local.

6.6.2 Évaluation du client BitTorrent

En comparaison d'autres clients BitTorrent libres, Transmission est plus lent en temps de téléchargement que rTorrent sur des réseaux à très haut débit ou sur un réseau local. Nous avons mesuré la bande passante à l'intérieur du *data center* de Tetaneutral. Nous avons lancé le transfert BitTorrent d'un DCP depuis un serveur physique vers une machine virtuelle. Avec Transmission, nous avons mesuré un débit maximal de 96 Mb/s. Avec le même DCP, nous avons mesuré, en utilisant rTorrent, des débits de 230 Mb/s. Le transfert se faisant sur le VPN Tinc, c'est donc le tunnel qui limite la bande passante de rTorrent.

En inspectant le code source de Transmission, nous avons relevé des boucles d'attente sur des *timers*, qui expliquent pourquoi la bande passante est plus faible. Ces boucles étant enfouies dans le code, il n'est pas facile de les modifier sans provoquer des dysfonctionnements dans le client BitTorrent.

À l'avenir, nous intégrerons l'interface utilisateur du client BitTorrent dans l'application Tuco. Nous pourrions ainsi plus facilement changer le client BitTorrent sans modifier les usages de l'exploitant. Nous étudierons s'il est plus pertinent d'utiliser rTorrent ou encore Deluge. L'avantage de Transmission est qu'il possède une interface RPC très complète entre l'interface et le client, et donc facilement intégrable dans une autre application.

Partant de ce constat, nous pouvons donc dire que, pour un exploitant, la bande passante maximale est de 96 Mb/s, ce qui correspond à un temps de transfert de 6 heures au mieux pour un DCP de 200 Go.

6.6.3 Évaluation globale de The DCP Bay

Les données enregistrées par le traqueur comportent les dates de début et de fin du téléchargement d'un DCP sur une machine. La table 6.4 donne un échantillon de transfert de DCP sur une fenêtre temporelle de deux jours et dix-huit heures.

Tous les DCP présentés dans cette table ont été initialement copiés sur une machine du réseau de l'IRIT (*utopian7*). Ils ont ensuite été transmis vers un serveur dans le *data center* de Tetaneutral (*tdcpb31*), puis vers les cinémas (*cineXX*). Dans ce cas précis, tous les DCP étaient disponibles pour tous les cinémas. Vu que le DCP est disponible au téléchargement sur *utopian7*, la machine *utopian7* constitue un *seeder* primaire. La machine *tdcpb31* est le serveur sur lequel tous les DCP sont enregistrés. C'est la machine principale, sur laquelle tous les DCP doivent finir par se trouver.

Dans la table 6.4, les DCP sont transférés vers cinq cinémas (appelés *cine35*, *cine69*, ...). Les cinémas observés ont tous des bandes passantes différentes. *Cine36* est connecté avec de la fibre optique à 100 Mb/s. Pour ce cinéma, le débit maximal atteint est de 21,30 Mb/s, soit cinq fois moins que le débit théorique annoncé par le FAI. Cela s'explique, au moins partiellement, par la qualité de service de la connexion réseau déployée par l'opérateur : en transfert réseau direct (sans le VPN Tinc), nous n'avons pas réussi à atteindre un débit supérieur à 80 Mb/s. De plus, le cinéma a téléchargé en même temps plusieurs DCP (*Timbuktu* et *Mr Turner*), ce qui a donc partagé sa bande passante descendante. La figure 3.5 comporte des références à des scripts LUA (langage de script utilisé par VLC). Avant de commencer le développement du module, nous avons testé la lecture de DCP à partir de ce langage. Cela aurait permis de lire des DCP sans avoir à créer un module spécifique. Cette solution s'est rapidement avérée trop contraignante, car elle ne nous permettait pas de gérer le chiffrement des DCP, ni la lecture de plusieurs bobines virtuelles.

Le serveur *tdcpb31* est connecté à un réseau à haut débit (avec une carte réseau de type Gigabit). La machine transfère dans les deux sens : elle charge depuis *utopian7* et envoie aussi vers les autres salles. Étant donné que l'on est sur un réseau pair à pair, le serveur envoie des

données pour un DCP vers une salle, alors qu'il n'est pas complètement arrivé sur la machine. Les débits de réception sur tdcpb31 sont lents (11 Mb/s) au regard de la bande passante attendue. Ce résultat s'explique car tdcpb31 n'a qu'un seul *seeder* : utopian7. Si, sur une autre machine dans le *data center* de Tetaneutral, on rajoutait aussi un client BitTorrent téléchargeant le même contenu que sur tdcpb31 depuis utopian7, on atteindrait des débits de 40 Mb/s sur tdcpb31 au lieu de 11 Mb/s. De plus, on arrive à atteindre un débit encore meilleur (60 Mb/s) sur tdpb31 quand on bloque son trafic sortant. On constate alors des limitations dans le tunnel du VPN, quand il y a simultanément du trafic entrant et du trafic sortant.

6.7 Vers une optimisation du CDN The DCP Bay

6.7.1 Analyse de l'existant

Pour chaque cinéma, nous devons distribuer un ou plusieurs DCP avant la première projection. Par ailleurs, un même DCP est souvent distribué à plusieurs exploitants. La solution que nous avons proposée est expérimentale. Il faudrait trouver une solution qui reste optimale malgré l'augmentation inévitable du trafic due au nombre croissant de salles à connecter et au nombre croissant de DCP à transférer. Outre le problème de la limitation de la bande passante due à l'usage du tunnel dans le VPN, qui doit plutôt être traité au niveau du réseau, il nous faut évaluer si notre système peut distribuer tous les DCP à tous les cinémas en respectant les délais (livraison du film avant la première projection) et en minimisant les coûts de bande passante.

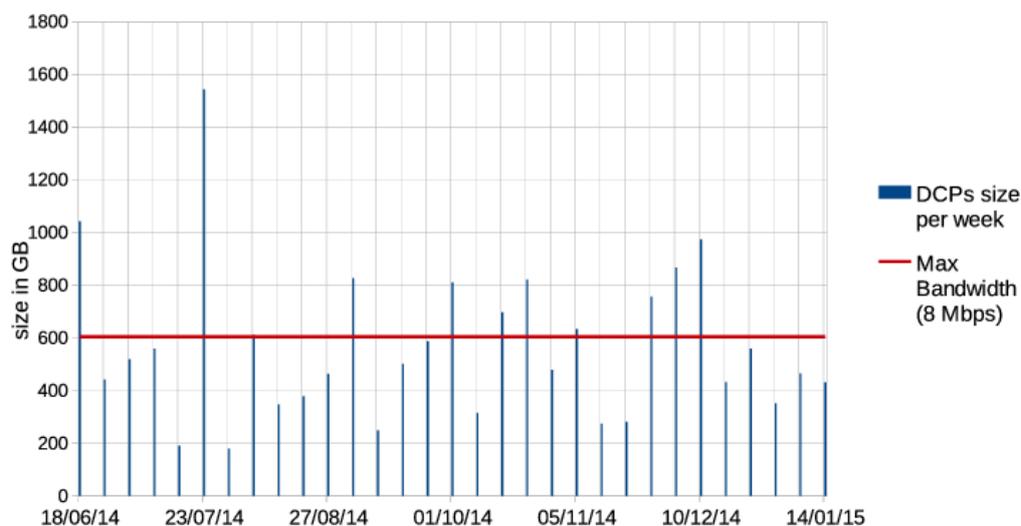


FIGURE 6.7 – Distribution de films au cinéma Utopia de Bordeaux. Huit mois d'exploitation de films sont représentés en situation réelle. Chaque semaine, de nouveaux films sont projetés. Les barres bleues représentent la quantité (en Gb) de DCP à recevoir.

TABLE 6.4 – Échantillon de transferts de DCP.

Machine	Début	Fin	Durée	Film	Taille (Go)	Débit (Mb/s)
cine35	2014-12-04 12:49:43	2014-12-06 16:19:13	2 jours, 3:29:30	<i>Men women</i>	202,34	8,73
tdcpb1	2014-12-04 15:46:32	2014-12-06 15:01:07	1 jour, 23:14:35	<i>Quand vient la nuit</i>	186,73	8,78
cine69	2014-12-04 21:45:31	2014-12-07 03:06:57	2 jours, 5:21:26	<i>White God</i>	203,16	8,46
tdcpb31	2014-12-05 11:07:25	2014-12-06 11:26:57	1 jour, 0:19:32	<i>Timbuktu</i>	127,80	11,68
tdcpb31	2014-12-05 12:56:53	2014-12-07 10:32:29	1 jour, 21:35:36	<i>Quand vient la nuit</i>	186,73	9,10
cine36	2014-12-05 13:46:24	2014-12-06 10:24:20	20:37:56	<i>Quand vient la nuit</i>	186,73	20,11
cine7	2014-12-05 13:48:46	2014-12-07 02:25:50	1 jour, 12:37:04	<i>Petite taupe</i>	70,12	4,26
tdcpb31	2014-12-05 15:05:26	2014-12-06 19:58:38	1 jour, 4:53:12	<i>Contes de la neige</i>	87,81	6,76
tdcpb31	2014-12-05 16:01:45	2014-12-08 00:47:30	2 jours, 8:45:45	<i>Mr Turner</i>	135,50	5,30
tdcpb1	2014-12-05 16:03:08	2014-12-07 19:21:49	2 jours, 3:18:41	<i>Mr Turner</i>	135,50	5,87
tdcpb1	2014-12-05 16:12:13	2014-12-07 11:07:45	1 jour, 18:55:32	<i>Timbuktu</i>	127,80	6,62
cine35	2014-12-05 18:33:33	2014-12-06 18:10:52	23:37:19	<i>Contes de la neige</i>	87,81	8,26
cine36	2014-12-05 19:48:15	2014-12-06 14:49:54	19:01:39	<i>Timbuktu</i>	127,80	14,93
cine36	2014-12-05 19:48:29	2014-12-06 09:59:48	14:11:19	<i>Contes de la neige</i>	87,81	13,75
cine36	2014-12-06 10:07:05	2014-12-07 00:15:25	14:08:20	<i>Mr Turner</i>	135,50	21,30
cine32	2014-12-06 17:53:34	2014-12-09 03:27:12	2 jours, 9:33:38	<i>A spell to...</i>	119,60	4,62
cine56	2014-12-06 23:23:40	2014-12-07 21:59:46	22:36:06	<i>Mr Turner</i>	135,50	13,32
cine36	2014-12-06 23:31:25	2014-12-08 04:26:47	1 jour, 4:55:22	<i>Tuer in homme</i>	114,39	8,79
cine34	2014-12-07 00:07:30	2014-12-08 09:42:40	1 jour, 9:35:10	<i>Mr Turner</i>	135,50	8,97
cine34	2014-12-07 00:07:45	2014-12-09 01:46:30	2 jours, 1:38:45	<i>Contes de la neige</i>	87,81	3,93
cine34	2014-12-07 00:07:56	2014-12-08 06:56:43	1 jour, 6:48:47	<i>Timbuktu</i>	127,80	9,22
cine34	2014-12-07 00:08:00	2014-12-10 08:32:47	3 jours, 8:24:47	<i>Men women</i>	202,34	5,59

La figure 6.7 représente huit mois de projection au cinéma Utopia de Bordeaux. Chaque mercredi, de nouveaux films sont projetés. Le nombre, et donc la taille, de nouveaux films à projeter varient d'une semaine à l'autre. Les barres bleues représentent la quantité (en Gb) de films à recevoir chaque semaine en situation réelle. La taille d'un DCP est très variable, car elle dépend des taux de compression et de la durée des films. En terme de transport de données, il est plus précis de parler de quantité de données plutôt que de nombre de films.

La ligne rouge indique la quantité de données (en Gb) qui peut être transportée par semaine par la connexion réseau du cinéma. Cette bande passante a été extraite des temps de transfert réels mesurés par le traqueur BitTorrent : elle est de 8 Mb/s.

Au cours des huit mois représentés sur la figure 6.7, il y a eu dix cas où nous n'avons pas pu envoyer en une semaine tous les DCP nécessaires à la première projection du mercredi. On peut, dans certains cas, utiliser la bande passante libre des semaines précédentes pour anticiper l'envoi des DCP, mais l'on ne connaît pas toujours à l'avance la disponibilité d'un DCP auprès du distributeur. En France, la sortie des nouveaux films est fixée au mercredi. Le distributeur ne peut mettre à disposition le DCP que peu de temps avant la sortie officielle du film. Il existe des cas où le film n'est disponible que quatre ou cinq jours avant la sortie du mercredi.

Une autre solution consiste à augmenter la bande passante de l'accès Internet du cinéma. Cela peut être fait en ajoutant une ou plusieurs lignes réseau, et en appliquant la méthode d'agrégation de lignes qui a déjà été présentée. Ajouter une ligne n'est pas toujours possible, en raison de la localisation ou de l'infrastructure du bâtiment accueillant le cinéma.

Pour The DCP Bay, notre but est de distribuer tous les DCP vers un cinéma. Si nous ne pouvons pas assurer tous les transferts sous forme dématérialisée, nous enverrons les DCP restants sur disque dur par transporteur physique, ce qui fera augmenter nos coûts de fonctionnement. Une solution simple consiste à envoyer le DCP de plus grande taille par transporteur physique, afin de libérer le plus possible de bande passante sur le réseau.

6.7.2 Contraintes sur l'optimisation de la distribution

Nous avons mesuré que le débit d'une machine *seeder* en *data center* ne pouvait pas dépasser 100 Mb/s. Les machines *seeders* en *data center* dans le modèle du CDN sont des SS. Nous disposons par défaut de deux machines en SS : *tdcpb31* et *mirror*. Avec ces deux machines, nous allouons une bande passante de 200 Mb/s.

Nous pouvons rajouter des machines agissant comme miroirs supplémentaires, même si cela génère un surcoût du fonctionnement opérationnel. Cela permet d'ajouter de la bande passante en sortie des serveurs, si la demande de téléchargement simultané d'un ou de plusieurs DCP est très forte. Nous devons limiter ce surcoût, tout en respectant les délais de téléchargement.

Nous ne maîtrisons pas la bande passante de réception des salles de cinéma. Nous pouvons être amenés à déclencher un envoi par transporteur physique, si nous constatons que la durée maximale de téléchargement risque d'être dépassée. Envoyer un DCP de cette manière génère aussi un surcoût qu'il faut minimiser.

6.8 Bilan

Nous avons proposé un CDN pour le transfert de DCP, opérationnel depuis plus d'un an, qui possède les caractéristiques d'un CDN classique (organisation hiérarchique des serveurs d'origine vers les salles de cinéma), mais aussi celles du fonctionnement en pair à pair (la distribution des contenus est réalisée par le protocole BitTorrent).

Ce système de distribution comporte une limitation de « bas niveau » liée à l'usage du VPN, qui n'est pas bloquante pour la distribution de DCP. En raison de cette limitation, nous savons que la bande passante maximale d'un serveur en *data center* est 100 Mb/s.

En ajoutant des *seeders* en *data center* (des SS, dans la dénomination d'un CDN), on augmente la bande passante d'autant. Néanmoins, l'ajout de *seeders* provoque un surcoût de fonctionnement. The DCP Bay dispose de trois *seeders* opérationnels en permanence. Pour ajouter un *seeder*, il suffit de dupliquer par BitTorrent le contenu sur un serveur dans un autre *data center*. La synchronisation de contenu entre *data centers* étant rapide, on peut ainsi augmenter la bande passante et répondre à un pic de demande pour un ou plusieurs DCP.

Avec The DCP Bay, en raison de l'utilisation du protocole BitTorrent au travers du VPN Tinc, nous avons donc mis en œuvre un système de distribution ouvert, interopérable et sécurisé.

Chapitre 7

Conclusion et perspectives

L'objectif de cette thèse était double : proposer, d'une part, un système de projection adapté au cinéma d'Art et Essai, et d'autre part, une solution de distribution de DCP.

vlc4dcp Nous avons développé un lecteur de DCP pour le cinéma numérique, qui est la pièce maîtresse du système de projection `vlc4dcp`. Pour réaliser ce lecteur, nous avons écrit un module dans VLC permettant de gérer le format DCP, c'est-à-dire de lire de manière synchronisée les images, le son et les sous-titres. Nous avons également montré comment lire les DCP chiffrés. Le module DCP a été officiellement publié dans VLC. Un problème de compatibilité de licences libres a été rencontré : nous utilisons la bibliothèque `ASDCPlib` pour gérer le conteneur MXF, qui est sous licence « New BSD license », VLC étant sous licence GPLv2. Ces deux licences ne sont pas compatibles, ce qui pose des problèmes de distribution du module DCP dans VLC.

Pour pouvoir lire un DCP à une fréquence d'au moins 24 images par seconde, il nous a fallu développer un décodeur JPEG2000 dédié. Ce décodeur a été écrit pour la bibliothèque multimédia `libavcodec`. Il est spécialisé pour la décompression des flux d'images JPEG2000 du cinéma numérique. Nous avons atteint, et même dépassé, la fréquence de 24 images par seconde au format 2K, ce qui n'était pas possible avec les solutions libres existantes. Nous avons utilisé le *multithreading* et des jeux d'instructions SSE, complétés par une solution de décompression avec pertes, afin d'atteindre les performances espérées. Le code développé pour le décodeur JPEG2000 a été partiellement publié dans `libav` et `ffmpeg` (les deux plateformes multimédia qui gèrent `libavcodec`).

Le décodage JPEG2000 à une fréquence d'au moins 24 images par seconde au format 2K a été atteint. Néanmoins, nous avons rencontré des difficultés à lire un DCP à cette fréquence et à cette résolution, à cause des traitements qui sont effectués après décompression. Nous avons tout de même réussi à atteindre cet objectif par interpolation de la résolution 1K. La qualité d'image est moindre, mais reste tout à fait acceptable si la projection est effectuée sur un écran de largeur inférieure à six mètres.

Nous avons donc démontré qu'il était possible de projeter des DCP en salle grâce à une solution en logiciel libre. Le système de projection proposé ne respecte pas toutes les exigences du DCI, notamment en matière de sécurité, mais il est bien moins coûteux (environ 10 000 €) que du matériel de projection certifié par le DCI (à partir de 50 000 €). À ce jour, le CNC exige que tout matériel de projection dans les salles de cinéma soit conforme au standard DCI. Lors du prochain renouvellement des projecteurs numériques, il faudra se demander si la solution que nous proposons n'est pas mieux adaptée aux « petits » cinémas, bien qu'elle ne soit que partiellement conforme à ce standard.

Pourquoi absolument chercher à jouer un DCP dans son format d'origine ? Ne pourrait-on pas le convertir dans un autre format avant de le projeter ? Il existe d'autres formats vidéo de très bonne qualité, dont les temps de décompression sont plus courts. En fait, une telle solution n'est pas envisageable car les DCP sont le plus souvent chiffrés. Et si l'on déchiffrait le DCP, une fois le KDM reçu, pour le convertir dans un autre format, cela constituerait une grave faille de sécurité vis-à-vis de la protection du contenu. On pourrait encore imaginer que le DCP soit chiffré avant d'être compressé, ce qui permettrait de le décompresser après réception, sans enfreindre la réglementation relative aux droits d'auteur. Mais cela ferait chuter les taux de compression, car le chiffrement réduit la corrélation : le chiffrement doit être effectué après compression. Cela poserait également aux exploitants des problèmes de stockage : il ne serait pas évident de stocker plusieurs films de 4 To. Et il ne serait pas non plus aisé de lire le fichier décompressé : pour projeter à une fréquence de 24 images par seconde des images non compressées de 12 Mo chacune, il faudrait pouvoir lire le disque à un débit de 288 Mo/s, ce qui est impossible avec un disque rotationnel. Les disques SSD permettent effectivement de lire à un tel débit, mais un disque SSD de 4 To coûte plus de 2000 €.

The DCP Bay Le deuxième objectif était de proposer un système de distribution de DCP neutre et interopérable, allant des distributeurs vers les exploitants. Le système que nous proposons est opérationnel et déjà utilisé pour transférer des DCP vers plus de 70 écrans. Nous avons développé un CDN pour le transport de DCP utilisant la technologie pair à pair, un protocole ouvert (le BitTorrent) et un VPN multi-point sous licence libre (Tinc). Les serveurs du système sont hébergés pas des FAI associatifs (FFDN). Le système est « interopérable », car il utilise des protocoles ouverts. Il est également « neutre », car il ne dépend ni d'un unique FAI, ni d'un d'installateur d'équipements, ni d'un distributeur, ni enfin de réseaux de salles de cinéma.

D'un point de vue économique, nous travaillons avec le GNCR à définir un tarif de transfert de DCP à l'unité (sans abonnement). Le tarif, qui doit être calculé au plus près du coût réel, permettra aux fournisseurs d'accès associatifs participant au projet de faire grandir leur structure. L'association Cinémascop, issue du GNCR, a lancé Indé-CP¹, qui est un projet de plateforme « commerciale » pour The DCP Bay.

Bilan Cette thèse a été effectuée dans le cadre d'une convention CIFRE entre l'IRIT et les cinémas Utopia (grâce à l'association ISF). Pour les cinémas Utopia, la collaboration avec un laboratoire d'informatique était une première. Les demandes d'Utopia au début de la thèse étaient précises : proposer un système de projection et de distribution sous licence libre. Lors du déroulement de la thèse, nous nous sommes concentrés sur ce double objectif, non sans difficultés, car il a été difficile de mener de front les deux volets. Nous avons commencé par travailler sur le système de projection. En milieu de thèse, nous nous sommes demandé s'il fallait approfondir le travail déjà effectué ou commencer à réfléchir au système de distribution. Pour Utopia, le système de distribution était un besoin à plus court terme que la projection. C'est la raison pour laquelle nous avons privilégié, pendant la deuxième moitié de la thèse, le système de distribution. Cela explique que cette thèse soit un travail exploratoire « en largeur », plutôt qu'une recherche « en profondeur » sur un problème très précis.

Perspectives Il nous reste à publier officiellement les optimisations logicielles du décodeur JPEG2000 que nous avons proposées dans libavcodec. Nous pourrions également continuer à optimiser le code pour pouvoir traiter le format HFR, ce qui nécessite d'atteindre des fréquences de décodage de 48 ou 60 images par seconde.

1. <http://www.cinemascope.net/inde-cp>

Il nous reste également à résoudre les problèmes d'affichage en résolution 2K. Il faudrait, au sein de VLC, optimiser l'affichage codé sur 12 bits. Nous pourrions également réduire la profondeur de couleur à 8 bits, mais il faudrait évaluer l'influence de ce choix sur la qualité des images projetées.

Le format DCP a été étendu à une nouvelle norme appelée *Interoperable Master Format* (IMF). L'IMF est un format de *master* numérique permettant de produire un contenu sous différents formats : DCP, Blu-ray, VOD... Comme le DCP, l'IMF utilise le JPEG2000, mais avec une profondeur de 16 bits par composante couleur.

Il faudra décider s'il est opportun de prendre en compte toutes les exigences du DCI en matière de sécurité, ou s'il vaut mieux s'en tenir à un système partiellement conforme à ces exigences. Puisque les systèmes de projection par laser commencent à être commercialisés, il faudrait vérifier si cette technologie ne pourrait pas répondre aux besoins des petites salles d'Art et Essai, surtout en matière de coût.

En ce qui concerne le système de distribution, The DCP Bay est voué à s'étendre à un plus grand nombre de distributeurs et de salles. Il nous faudra donc continuer à optimiser les coûts opérationnels de fonctionnement du réseau.

Nous souhaitons réaliser un système de distribution de DCP complètement pair à pair, mais les débits fournis par les FAI empêchent les cinémas d'avoir un débit élevé. Or, la taille moyenne d'un DCP est de 200 Go. Nous avons donc recours à des serveurs en *data center* pour pouvoir fournir la bande passante nécessaire au transfert de DCP. Pour éviter cela, il faudrait que les connexions par fibre optique se généralisent et couvrent tout le territoire (il existe des opérateurs spécialisés, capables de fournir des connexions à très haut débit, mais les frais d'installation et de fonctionnement sont prohibitifs pour une salle de cinéma).

Si tous les cinémas disposaient de débits de 100 ou 200 Mb/s symétriques (c'est-à-dire avec un débit montant égal au débit descendant), nous disposerions alors d'un système de distribution décentralisé « horizontal », par opposition au système de distribution actuel, qui est « vertical » : du laboratoire au distributeur, en passant par l'opérateur de transfert, jusqu'au cinéma. Nous permettrions ainsi à des exploitants d'être directement connectés, et de s'échanger du contenu sans passer par un opérateur tiers, voire même de se passer des machines en *data center*. Mais bien sûr, les distributeurs et les exploitants pourraient continuer à gérer la programmation des salles.

Nous souhaitons, à travers ce projet, faire un pont entre les défenseurs de la neutralité de l'Internet et les défenseurs de la diversité cinématographique. Si, aujourd'hui, ce projet est centré autour du circuit cinématographique traditionnel, il offre une perspective d'avenir pour la diffusion de tous types d'œuvres, et donc pour la promotion de la diversité des contenus, afin d'éviter la focalisation sur les œuvres les plus rentables.

L'utilisation combinée de vlc4dcp et de The DCP Bay pourrait permettre non seulement le renouveau des salles de quartier et des ciné-clubs, mais également la création de lieux de projection alternatifs (indépendants du CNC) qui pourraient, avec l'accord des distributeurs, recevoir et projeter des DCP.



Annexe A

Acronymes

ADSL	<i>Asymmetric Digital Subscriber Line</i>	99
AES	<i>Advanced Encryption Standard</i>	28
AFCAE	Association Française des Cinémas d'Art et d'Essai	18
CBC	<i>Cipher Block Chaining</i>	79
CDN	<i>Content Delivery Network</i>	15
CE	<i>Customer Edge</i>	95
CIFRE	Conventions Industrielles de Formation par la REcherche	14
CNC	Centre National du Cinéma et de l'image animée	17
CPL	<i>Composition Play List</i>	29
CST	Commission Supérieure et Technique de l'image et du son	36
DCDM	<i>Digital Cinema Distribution Master</i>	25
DCI	<i>Digital Cinema Initiatives</i>	14
DCP	<i>Digital Cinema Package</i>	25
DCSS	<i>Digital Cinema System Specification</i>	14
DKDM	<i>Distribution KDM</i>	84
DLP	<i>Digital Light Processing</i>	13
DM	<i>Decision Maker</i>	92
DRM	<i>Digital Rights Management</i>	80
DWT	<i>Discrete Wavelet Transform</i>	62
EBCOT	<i>Embedded Block Coding with Optimized Truncation</i>	65
ES	<i>Elementary Stream</i>	51
ETM	<i>Extra-Theater Messages</i>	83
FA	Film-Annonce	102
FAI	Fournisseur d'Accès à Internet	42
FFDN	Fédération FDN	42
FIPS	<i>Federal Information Processing Standards</i>	80
FPGA	<i>Field-Programmable Gate Array</i>	69
FTP	<i>File Transfer Protocol</i>	32
GLSL	<i>OpenGL Shading Language</i>	55
GNCR	Groupement National des Cinémas de Recherche	18

GOP	<i>Group Of Pictures</i>	70
GPGPU	<i>General-purpose Processing on Graphics Processing Units</i>	70
GPL	<i>GNU General Public License</i>	56
HDCP	<i>High-bandwidth Digital Content Protection</i>	38
HDMI	<i>High Definition Multimedia Interface</i>	38
HFR	<i>High Frame Rate</i>	26
HVMT	<i>Human Vision Modulation Threshold</i>	26
IDWT	<i>Inverse Discrete Wavelet Transtorm</i>	73
ILA	<i>Image Light Amplifier</i>	13
IMB	<i>Integrated Media Block</i>	32
IMCT	<i>Inverse Multi-Component Transform</i>	73
IMF	<i>Interoperable Master Format</i>	117
ISDCF	<i>Inter-Society Digital Cinema Forum</i>	30
ISF	<i>Indépendants, Solidaires et Fédérés</i>	18
ISO	<i>International Organization for Standardization</i>	14
KDM	<i>Key Delivery Message</i>	28
KLV	<i>Key-Length-Value</i>	28
MB	<i>Media Block</i>	32
MXF	<i>Material eXchange Format</i>	28
OAEP	<i>Optimal Asymmetric Encryption Padding</i>	81
OCAP	<i>Open CAPtion</i>	103
OS	<i>Origin Server</i>	92
PE	<i>Provider Edge</i>	95
PKCS	<i>Public Key Cryptographic Standards</i>	82
PKL	<i>PacKing List</i>	29
PNG	<i>Portable Network Graphics</i>	26
PSNR	<i>Peak Signal to Noise Ratio</i>	76
PTS	<i>Presentation Time Stamp</i>	48
RFC	<i>Request For Comment</i>	55
RPC	<i>Remote Procedure Call</i>	102
RSA	<i>Rivest-Shamir-Adleman</i>	28
SM	<i>Security Manager</i>	80
SMPTE	<i>Society of Motion Picture and Television Engineers</i>	13
SMS	<i>Screen Management System</i>	32
SMT	<i>Simultaneous MultiThreading</i>	73
SPDIF	<i>Sony/Philips Digital InterFace</i>	38
SS	<i>Surrogate Server</i>	92
SSD	<i>Solid-State Drive</i>	39
SSE	<i>Streaming SIMD Extensions</i>	70
SSIM	<i>Structural SIMilarity</i>	77
SXRD	<i>Silicon X-tal Reflective Display</i>	33
TCP	<i>Transmission Control Protocol</i>	95
TDL	<i>Trusted Device List</i>	80

TIFF	<i>Tagged Image File Format</i>	25
TLS	<i>Transport Layer Security</i>	80
TMS	<i>Theater Management System</i>	32
UUID	<i>Universally Unique Identifier</i>	29
VLC	<i>VideoLAN Client</i>	15
VOB	<i>Video Object</i>	51
VPF	<i>Virtual Print Fee</i>	21
VPN	<i>Virtual Private Network</i>	42
VT	<i>Visibility Threshold</i>	71
WAV	<i>Waveform Audio-file Format</i>	26
XML	<i>eXtensible Markup Language</i>	26

Bibliographie

- [1] OpenJPEG – An open-source JPEG 2000 codec written in C. <http://www.openjpeg.org/>. 70, 71
- [2] The European Cinema Yearbook, Media Salles. <http://www.mediasalles.it/yearbook.htm>. 21
- [3] M. D. ADAMS et F. KOSSENTINI : JASPER: A Software-Based JPEG-2000 Codec Implementation. In *Proceedings of the International Conference on Image Processing – ICIP 2000*, volume 2, pages 53–56, 2000. 70
- [4] C. ANGELOPOULOS : On opportunities and challenges for european cinema in the digital era. Technical Report, European Commission, septembre 2010. 22, 23, 38, 79
- [5] M. ANTONINI, M. BARLAUD, P. MATHIEU et I. DAUBECHIES : Image coding using wavelet transform. *IEEE Transactions on Image Processing*, 1(2):205–220, 1992. 62
- [6] P. G. J. BARTEN : *Contrast Sensitivity of the Human Eye and Its Effects on Image Quality*. Press Monographs. SPIE Optical Engineering Press, 1999. 26
- [7] A. BILGIN et M. W. MARCELLIN : JPEG2000 for digital cinema. In *Proceedings of the 2006 IEEE International Symposium on Circuits and Systems – ISCAS 2006*, pages 3878–3881. IEEE, mai 2006. 27, 67
- [8] N. BLEFARI-MELAZZI, D. DI SORTE, M. FEMMINELLA, L. PIACENTINI et G. REALI : Performance evaluation of a multicast-based solution for wireless resources discovery. In *Proceedings of the IEEE International Conference on Communications*, volume 5, pages 3254–3260, mai 2005. 43
- [9] J. A. BLOOM : Digital Cinema Content Security and the DCI. In *Proceedings of the 40th Annual Conference on Information Sciences and Systems*, pages 1176–1181, mars 2006. 79
- [10] J. A. BLOOM : Security in Digital Cinema. In *Multimedia Security Technologies for Digital Rights Management*, pages 413–432. Academic Press, 2006. 80
- [11] V. BRUNS et H. SPARENBERG : Comparison of Code-Pass-Skipping Strategies for Accelerating a JPEG 2000 Decoder. In *Proceedings of the ITG-Fachtagung für Elektronische Medien (Fernsehseminar)*, mars 2013. 71
- [12] V. BRUNS, H. SPARENBERG et S. FOSSEL : Video Decoder and Methods for Decoding a Sequence of Pictures, patent WO 2012004164 A1, janvier 2012. 70
- [13] S. BYERS, L. FAITH CRANOR, E. CRONIN, D. KORMAN et P. MCDANIEL : An analysis of security vulnerabilities in the movie production and distribution process. *Telecommunications Policy*, 28(7-8):619–644, 2004. A selection of papers from the 31st Annual Telecommunications Policy Research Conference. 81
- [14] B. COHEN : Incentives build robustness in BitTorrent. In *Proceedings of the Workshop on Economics of Peer-to-Peer Systems*, volume 6, pages 68–72, 2003. 93

- [15] DCI : *Digital Cinema System Specification*. Digital Cinema Initiatives, 1.2 édition, mars 2006. 25
- [16] DCI : *Digital Cinema System Specification*. Digital Cinema Initiatives, 1.2 with errata édition, mars 2012. 24, 33, 39, 43, 80
- [17] A. DESCAMPE, F.-O. DEVAUX, G. ROUVROY, J.-D. LEGAT, J.-J. QUISQUATER et B. MACQ : A flexible hardware JPEG2000 decoder for digital cinema. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(11):1397–1410, 2006. 69
- [18] D. DI SORTE, M. FEMMINELLA, G. REALI et L. ROSATI : Definition and performance evaluation of a request routing algorithm to distribute digital cinema contents. *In Proceedings of the 4th International Telecommunication Networking Workshop on QoS in Multiservice IP Networks*, pages 27–32, février 2008. 94, 95, 96
- [19] J. FARCHY : *Et pourtant ils tournent ... Économie du cinéma à l'ère du numérique*. Medias Essais. INA éditions, 2011. 19, 20, 21, 24, 31
- [20] ISO : *ISO/IEC 15444-1 – Information technology – JPEG 2000 image coding system: Core coding system – Part 1*, août 2004. 61, 62, 68
- [21] L. JIMENEZ-RODRIGUEZ, F. AULI-LLINAS et M. W. MARCELLIN : Visually lossless strategies to decode and transmit JPEG2000 imagery. *IEEE Signal Processing Letters*, 21(1):35–38, 2014. 71, 77
- [22] G. KENNEL : *Color and Mastering for Digital Cinema*. Digital cinema industry handbook series. Taylor & Francis, 2012. 26
- [23] A.-M. KHAN PATHAN et R. BUYYA : A taxonomy and survey of content delivery networks. Technical report, Grid Computing and Distributed Systems Laboratory, University of Melbourne, 2007. 92
- [24] S. KHANVILKAR et A. KHOKHAR : Experimental evaluations of open-source Linux-based VPN solutions. *In Proceedings of the 13th International Conference on Computer, Communications and Networks*, pages 181–186, octobre 2004. 104
- [25] RSA LABORATORIES : *Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1*, 2003. 82
- [26] R. LE, I. R. BAHAR et J. L. MUNDY : A novel parallel Tier-1 coder for JPEG2000 using GPUs. *In Proceedings of the 9th IEEE Symposium on Applications Specific Processors – SASP 2011*, pages 129–136. 70, 71
- [27] R. LE, J. L. MUNDY et R. I. BAHAR : High Performance Parallel JPEG2000 Streaming Decoder Using GPGPU-CPU Heterogeneous System. *In Proceedings of the 23rd IEEE International Conference on Application-specific Systems, Architectures and Processors – ASAP 2012*, pages 16–23. 70, 71
- [28] M.-J. LEE, K.-S. KIM et H.-K. LEE : Digital Cinema Watermarking for Estimating the Position of the Pirate. *IEEE Transactions on Multimedia*, 12(7):605–621, novembre 2010. 80
- [29] P. LEVRIER : Rapport du groupe de travail sur le modèle économique du cinéma numérique. Rapport technique, CNC, avril 2008. 19, 20
- [30] J. MAS RIBÉS, S. BAUDRY, P. CORREA, D. DOYEN, B. MACQ, Y. MAETZ, O. DE MARNEFFE, P. NGUYEN, J.-F. NIVART et I. WOLF : *European Digital Cinema Security Whitebook*. Presses Universitaires de Louvain, 2007. 19, 21, 80
- [31] J. MATELA, M. ŠROM et P. HOLUB : Low GPU Occupancy Approach to Fast Arithmetic Coding in JPEG2000. *In LNCS 7119*, pages 136–145. 2012. 70, 71

- [32] P. MICANTI, G. BARUFFA et F. FRESCURA : Reliable D-Cinema Multicasting over Heterogeneous Networks. *Journal of Communications*, 2(5), 2007. 31
- [33] P. MICANTI, F. FRESCURA et G. BARUFFA : Digital Cinema package transmission over wireless IP networks. *In Proceedings of the IEEE International Symposium on Wireless Communication Systems – ISWCS’08*, pages 154–158, octobre 2008. 28
- [34] NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY : *Advanced Encryption Standard (AES) – FIPS 197*, 2001. 79
- [35] R. L. RIVEST, A. SHAMIR et L. ADLEMAN : A Method for Obtaining Digital Signatures and Public-key Cryptosystems. *Communications of the ACM*, 21(2):120–126, février 1978. 81
- [36] P. RODRIGUEZ et E. W. BIERSACK : Dynamic Parallel Access to Replicated Content in the Internet. *IEEE/ACM Transactions on Networking*, 10(4):455–465, août 2002. 95
- [37] G. ROUVROY, F.-X. STANDAERT, F. LEFÈBVRE, J.-J. QUISQUATER, B. MACQ et J.-D. LEGAT : Reconfigurable hardware solutions for the digital rights management of digital cinema. *In Proceedings of the 4th ACM Workshop on Digital Rights Management – DRM’04*, pages 40–53, 2004. 27, 69
- [38] A. SKODRAS, C. CHRISTOPOULOS et T. EBRAHIMI : The JPEG2000 still image compression standard. *IEEE Signal Processing Magazine*, 18(5):36–58, septembre 2001. 65
- [39] G. SLIEPEN : The difficulties of a peer-to-peer VPN on the hostile Internet. *In Proceedings of the Free and Open Source Software Developers’ European Meeting - FOSDEM*, 2010. 109
- [40] SMPTE : *D-Cinema Operations – Digital Certificate – SMPTE 430-2-2006*, 2006. 83
- [41] SMPTE : *D-Cinema Operations – Key Delivery Message – SMPTE 430-1-2006*, 2006. 83, 84, 86
- [42] SMPTE : *D-Cinema Packaging – MXF Track File Essence Encryption – SMPTE 429-6-2006*, 2006. 79
- [43] SMPTE : *EG 432-1-2007 – Digital Source Processing – Color Processing for D-Cinema*, 2007. 26
- [44] SMPTE : *D-Cinema Distribution Master – Closed Caption and Closed Subtitle – SMPTE 428-10-2008*, 2008. 54
- [45] SMPTE : *D-Cinema Operations – Generic Extra-Theater Message Format – SMPTE 430-3-2008*, 2008. 81, 82, 83
- [46] H. SPARENBERG, T. JOORMANN, C. FELDHEIM et S. FOESSEL : Adaptive RAID: Introduction of optimized storage techniques for scalable media. *In Proceedings of the 20th IEEE International Conference on Image Processing*, pages 1826–1830, septembre 2013. 43
- [47] D. TAUBMAN : Multithreaded processing paradigms for JPEG2000. *In Proceedings of the 14th IEEE International Workshop on Multimedia Signal Processing – MMSP 2012*, pages 164–169. 70, 71
- [48] D. TAUBMAN : High performance scalable image compression with EBCOT. *IEEE Transactions on Image Processing*, 9(7):1158–1170, juillet 2000. 65
- [49] D. S. TAUBMAN et M. W. MARCELLIN : *JPEG2000: image compression fundamentals, standards, and practice*, volume 1 de *The Kluwer international series in engineering and computer science*. Kluwer Academic Publishers, 2002. 65
- [50] Z.-S. WANG, L. LI, X.-S. WANG, K. ZHANG, K. WANG, P. YAO, W.-D. CAO et H.-H. SHEN : A Digital Cinema Playback system compliant with the DCI specification. *In Proceedings of the Picture Coding Symposium – PCS 2009*, pages 1–4, mai 2009. 69