

Google prends les ARM pour défendre Theora

Comme vous le savez, HTML5 est la cinquième version du langage utilisé pour concevoir les pages Web, qui propose enfin aux développeurs le moyen d'afficher directement et simplement les vidéos dans la page, de la même façon qu'il est déjà possible de le faire pour des images. C'est à dire de façon simple, ouverte et puissante, sans la contrainte de recourir à un plugin propriétaire comme Flash.



Mais la taille des vidéos doit être réduite pour pouvoir être diffusées sur Internet, et comme il a déjà été dit dans ces colonnes, HTML5 est muet quant au format à utiliser pour compresser ces vidéos (on parle aussi de codec). Aujourd'hui, deux codecs peuvent techniquement assumer ce rôle clé : Theora et H264 (en attendant une possible libération du codec VP8 racheté par Google). Mais alors que le premier est entièrement libre et utilisable sans aucune restriction, le deuxième est breveté et son usage est rigoureusement encadré et soumis à redevance.

On comprend alors que l'affrontement de ces deux codecs n'est que la traduction de l'affrontement de deux visions : tandis que Theora doit permettre de maintenir un Web ouvert où chacun serait libre de publier du contenu ou de créer des applications innovantes, H264 permettra à quelques entreprises privées de faire main basse sur le Web.

À côté de l'avènement de HTML5, un autre facteur est en train de rebattre les cartes : il s'agit du nombre grandissant d'appareils mobiles reliés à Internet^[1].

Pour ces appareils mobiles, la question de la puissance processeur et de la consommation électrique se pose de manière nouvelle : ces appareils sacrifient souvent la première au profit de la seconde.

Ainsi, à l'inverse de ce qui se passe dans le monde des micro-ordinateurs de bureau, où l'architecture x86 développée par Intel domine, l'univers des appareils ultra mobiles est dominé par une architecture concurrente : ARM.

Les processeurs ARM (pour Advanced Risc Machine) sont des processeurs fabriqués sous licence par différentes sociétés de par le monde à partir des travaux de la société anglaise ARM. Cette société conçoit des processeurs consommant peu d'énergie qui se retrouvent ensuite dans la plupart des appareils mobiles : des sociétés comme Nokia, Google (Nexus One), HTC, Samsung, Apple (Iphone/iPad), Sony Ericsson, Palm (Plam Pre), Siemens, LG, ou encore Motorola l'utilisent dans leurs smartphones.

Dans la mesure où l'on retrouve souvent le codec H264 dans les vidéos en Flash et où le format H264 exige d'importantes ressources processeur pour être décodé, de nombreux appareils mobiles ont été optimisés pour aider au décodage du H264.

De fait, ce handicap initial est devenu paradoxalement un avantage pour le H264 : bien que Theora demande beaucoup moins de ressources pour être décodé et soit ainsi plus adapté aux appareils mobiles, certains lui opposent le fait qu'il ne bénéficie pas du même support matériel que le H264.

Heureusement la situation n'est pas figée et de nombreux développeurs sont à pied d'œuvre pour permettre à Theora de tirer parti de ces optimisations matérielles (en attendant que les fabricants implémentent eux-même leurs optimisations pour Theora).

C'est le cas notamment de Google qui investit dans le développement de theorARM, une version de Theora optimisée

pour les processeurs ARM, comme l'explique Robin Watts sur le blog « des projets Open Source » de Google dans le billet traduit ci-après.

Avant de laisser la parole à Robin Watts, remarquons que, depuis que Theora est pressenti comme codec possible pour la vidéo sur le Web, celui-ci s'améliore de manière constante. Tous les défauts relevés à l'encontre de Theora sont en passe d'être corrigés grâce aux efforts conjugués de toute une communauté de développeurs talentueux regroupée derrière la fondation Xiph.Org (qui développe Theora) dont Red Hat, les fondations à but non lucratif Wikimédia et Mozilla, et à présent Google.

H264, de son côté, reste irrémédiablement entravé par son péché originel : celui d'être breveté jusqu'à l'os. Et les meilleurs développeurs du monde n'y pourront rien changer.

Bref, là où Theora promet le meilleur, à tous points de vue, H264 n'annonce que le pire.

Ça bouge du côté de la vidéo sur le Web

Interesting times for Video on the Web

*Robin Watts – 9 avril 2010 – Google Open Source Blog
(Traduction Framalang : Olivier Rosseler et Goofy)*

Si je vous disais que Google a participé au financement d'une version du codec Theora optimisée pour les processeurs ARM, vous seriez nombreux à passer directement au billet suivant de ce blog. Personne ne se soucie des codecs audio ou vidéo, du moins tant qu'ils fonctionnent.

Demandez à la plupart des utilisateurs d'ordinateurs quel est leur codec favori et ils vous regarderont comme si vous aviez demandé leur moteur de machine à laver préféré. « On veut juste que ça marche ! » vous répondront-ils. C'est exactement la même chose pour ceux qui créent et publient du contenu sur

le Web. À chaque fois qu'une vidéo est inaccessible au visiteur d'un site Web, c'est un message qui n'atteint pas sa cible. Peu importe la qualité du message et le temps passé à le peaufiner, ou les techniques virales employées pour attirer les visiteurs ; si la page Web n'affiche qu'un beau rectangle vide à l'endroit du contenu, tous ces efforts auront été vains.

Publiez la vidéo afin qu'elle soit lisible partout

L'idée est bonne, mais sa mise en pratique est loin d'être simple. Il n'existe aucun standard vidéo actuellement sur le Web. Certains sites utilisent la technologie Flash, restreignant ainsi leur auditoire aux utilisateurs disposant d'un lecteur Flash et excluant de fait la plupart des téléphones. Certains se servent de lecteurs Java intégrés, mais l'auditoire se limite alors au seul public disposant de machines suffisamment puissantes pour décoder l'audio et la vidéo dans une machine virtuelle, ce qui exclut tout ce qui est plus lent qu'un ordinateur portable. D'autres encore se reposent sur les lecteurs natifs à telle plateforme (comme Windows Media Player) et se coupent ainsi de toutes les autres plateformes. D'autres enfin ne proposent que le lien vers la vidéo et s'en remettent au lecteur vidéo que l'utilisateur a pu éventuellement installer.

On est bien loin du « ça marche, tout simplement », et aucune de ces solutions ne semble en prendre le chemin. Que ça vous plaise ou non, avec la multiplication des appareils connectés, ce Graal va être de plus en plus difficile à atteindre. À une époque, il suffisait de s'assurer que la vidéo était lisible sur PC et sur Mac. Maintenant il faut jongler entre Android, ChromeOS, iPhoneOS, Linux, Maemo, Symbian et plein d'autres systèmes encore. Et ce n'est pas tout, puisqu'il faut prendre en compte toute une gamme de processeurs de puissances différentes, depuis les ordinateurs de bureaux aux téléphones en passant par les ordinateurs portables, les netbooks et les PDA. Le problème est loin d'être résolu, bien au contraire.

Heureusement, HTML5 nous apporte de l'espoir. Cette nouvelle version d'HTML (le langage de base utilisé pour écrire les pages Web) introduit un élément video.

Un standard existe maintenant pour que les gens spécifient dans le code de la page l'intégration d'une vidéo. À chaque navigateur, ensuite, d'implémenter la lecture des vidéos : soit qu'il lise lui-même la vidéo soit qu'il délègue cette tâche à un lecteur incorporé ou externe. Dans tous les cas, la décision appartient à l'utilisateur au lieu d'être imposée par le créateur du contenu. Le plus génial dans tout cela, c'est que cette fonction est déjà supportée : Firefox, Opera, Chrome et Safari prennent déjà en charge le HTML5 et les autres navigateurs suivront rapidement.

Alors, problème réglé ?

À vrai dire, non. Il existe désormais un langage commun pour publier des vidéos, c'est un bon début, mais la question du format n'est pas résolue. Là, il n'y a pas de « taille unique ». La vidéo est-elle visionnée sur un téléphone avec un petit écran, sur un netbook, sur un PC de bureau, ou sur un écran LED 3D de 150 pouces à la résolution quatre fois supérieure à celle de la TVHD ? La taille de l'écran, la connectique et la puissance de calcul jouent chacune un rôle ici. Un peu comme la transition de la VHS au DVD puis au BluRay pour les vidéos télévisées, la vidéo sur le Web va s'améliorer au fur et à mesure. Et c'est très bien ainsi : la technologie des serveurs Web permet d'adapter la vidéo au navigateur et à l'appareil utilisés.

En revanche, il nous faut assurément nous accorder sur un plus petit dénominateur commun, un format standard qui servira de format par défaut si tout le reste échoue. Ça ne doit pas nécessairement être le format le plus complexe, ou le plus marketé, ni même celui créé par l'union du plus grand nombre d'entreprises. Il doit simplement être disponible, partout. Le codec le plus adapté actuellement est Ogg Theora, dérivé du

codec VP3 libéré par On2 il y a quelques années. Il se débrouille plutôt bien en termes de qualité et de compression, et il n'a pas à rougir face à ses rivaux plus populaires comme le MPEG4, tout en étant bien plus simple à décoder. Mais surtout, il se démarque de ses concurrents par le fait qu'il est libre. Vraiment libre. Pas seulement « libre d'utilisation dans les décodeurs » ou « libre d'utilisation si vous signez cet accord de licence compliqué », mais vraiment, honnêtement, authentiquement, 100% libre. Les spécifications du flux et le code source de l'encodeur/décodeur sont accessibles publiquement et peuvent être librement utilisées/modifiées par n'importe qui. Theora a toujours été développé avec comme objectif premier d'éviter les brevets. Aucun autre codec que Theora ne peut se targuer de n'enfreindre aucun brevet et de n'être soumis à aucune redevance tout en tenant malgré tout la dragée haute à ses concurrents.

Alors, qu'est-ce qu'on attend ?

Décoder une vidéo demande beaucoup de ressources au processeur. Un important travail reste à accomplir avant de réaliser le rêve de lire une vidéo sur n'importe quel appareil. Theora est bien moins complexe que bon nombre de ses concurrents : les autres codecs nécessitent souvent du matériel dédié pour atteindre des performances décentes, alors que, bien programmé, Theora peut être décodé sans aide extérieure. En fait, sur les ordinateurs portables ou de bureau, le décodage à la volée est possible dans un lecteur Java intégré (comme par exemple l'excellent Cortado, lui aussi libre), ce qui permet la lecture de vidéos dans les navigateurs ne prenant pas encore en compte l'élément video. Mais ça n'est pas envisageable pour les appareils comme les PDA, les téléphones, les netbooks, les tablettes ou les lecteurs portables. Ces engins reposent souvent sur des processeurs ARM qui sont bien moins énergivores que les processeurs qui équipent habituellement les micro-ordinateurs. Malgré les gains de puissance continus réalisés au cours de

ces dernières années, ils ne rivalisent pas encore avec leurs cousins plus costauds pour ce qui est de la puissance pure. Cette gamme très importante d'appareils construits autour de processeurs ARM représente le dernier défi pour un décodage fluide de Theora. Tout progrès d'efficacité réalisé se traduit par davantage d'autonomie ou des écrans plus grands.

C'est là que l'aide financière de Google intervient. En subventionnant le développement de TheorARM (une version libre de Theora optimisée pour les processeurs ARM), ils nous aident à nous rapprocher du jour où la vidéo fonctionnera partout sur le Web, pour tout le monde. Ça serait plutôt pas mal, non ? Ça y est, maintenant vous pouvez passer au billet suivant du blog.

Notes

[1] Crédit photo : Titlap (Creative Commons By)