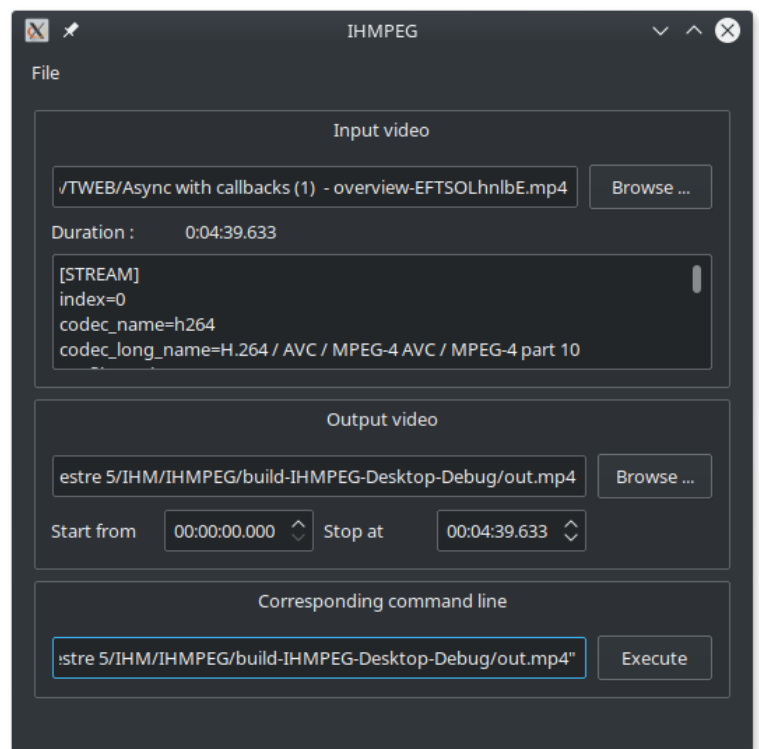
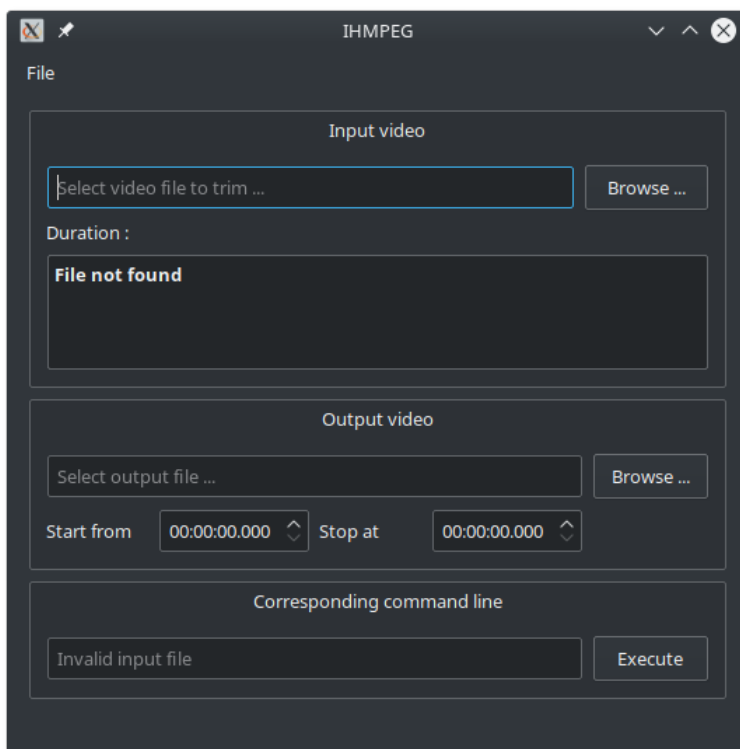


IHMPEG

Interface

Nous avons choisi de disposer les éléments d'interface verticalement, dans l'ordre chronologique de leur utilisation normale : input => output => résultat. Sur les captures suivantes, l'interface utilise un thème sombre car l'environnement graphique (Plasma) utilisé par la machine qui a servi d'exemple utilise son propre thème sombre pour toutes les applications Qt. La première capture illustre un input invalide, la seconde un input valide.



Input video

Le bouton « Browse ... » ouvre l'explorateur de fichiers en mode ouverture. A la selection d'un fichier, son chemin d'accès est entré dans la champs de gauche. Ce champs peut être édité manuellement.

La durée de la vidéo et les informations détaillées (fournies par ffprobe) s'affichent en-dessous. Ces informations sont mises à jour à chaque modification du chemin d'accès du fichier source.

Si le fichier n'existe pas, est un dossier ou n'est pas un fichier vidéo, un message d'erreur adapté est affiché en **gras** à la place des informations de la vidéo.

Output video

De nouveau, le bouton « Browse » ouvre l'explorateur de fichiers en mode sauvegarde. Si le fichier existe, il demande confirmation à l'utilisateur pour l'écrasement du fichier existant (l'exécution de la commande lors de la dernière étape de demande pas cette confirmation une deuxième fois).

Les champs « Start from » et « Stop at » sont préremplis afin de correspondre à la durée totale de la vidéo.

Corresponding command line

Si le chemin d'accès spécifié n'existe pas, si les temps de début et de fin sont invalides, ou si le fichier source est invalide, un message d'erreur grisé s'affiche à la place de la ligne de commande. Appuyer sur le bouton « Execute » affiche ce message d'erreur dans un popup pour une meilleure visibilité.

Si tout est valide, la ligne de commande permettant de lancer le traitement de la vidéo est affichée, et le bouton « Execute » lance la commande. A la fin du traitement, un popup signale le succès de l'opération s'affiche.

Problèmes et solutions

La seule difficulté rencontrée était la gestion des mauvaises entrées de l'utilisateur. Il y a pour nous quatre cas de figure principaux, découpés en sous-cas :

1. Le fichier source est invalide.
 - a) Il est introuvable.
 - b) Ce n'est pas un fichier vidéo.
2. Le chemin d'accès du fichier de sortie n'existe pas.
3. Le temps de début de la vidéo en sortie excède le temps de fin (overlap).
4. Le temps de fin de la vidéo en sortie excède la durée de la vidéo source.

S'il y a une mauvaise entrée de l'utilisateur, le champs « Corresponding command line » indique de quel problème principal (1, 2, 3 ou 4) il s'agit. S'il y a plusieurs entrées invalides, c'est la première qui est signalée.

Lorsque le fichier source est invalide, l'utilisateur est informé (par le champs d'informations de la vidéo) s'il s'agit du sous-problème a) ou b).

Conclusion

Dans l'ensemble, il s'agit d'une application ordinaire qui pose peu de défis du point de vue de l'interface. Avec notre expérience d'utilisateur, il a suffi de prendre exemple sur les interfaces basiques dont on a l'habitude.