
Towards a distributed implementation of a data extraction system for monitoring the processing of video clips

Auteur : Vasbinder, Thomas

Promoteur(s) : Wolper, Pierre

Faculté : Faculté des Sciences appliquées

Diplôme : Master en ingénieur civil en informatique, à finalité approfondie

Année académique : 2015-2016

URI/URL : <http://hdl.handle.net/2268.2/1306>

Avertissement à l'attention des usagers :

Tous les documents placés en accès ouvert sur le site le site MatheO sont protégés par le droit d'auteur. Conformément aux principes énoncés par la "Budapest Open Access Initiative"(BOAI, 2002), l'utilisateur du site peut lire, télécharger, copier, transmettre, imprimer, chercher ou faire un lien vers le texte intégral de ces documents, les disséquer pour les indexer, s'en servir de données pour un logiciel, ou s'en servir à toute autre fin légale (ou prévue par la réglementation relative au droit d'auteur). Toute utilisation du document à des fins commerciales est strictement interdite.

Par ailleurs, l'utilisateur s'engage à respecter les droits moraux de l'auteur, principalement le droit à l'intégrité de l'oeuvre et le droit de paternité et ce dans toute utilisation que l'utilisateur entreprend. Ainsi, à titre d'exemple, lorsqu'il reproduira un document par extrait ou dans son intégralité, l'utilisateur citera de manière complète les sources telles que mentionnées ci-dessus. Toute utilisation non explicitement autorisée ci-avant (telle que par exemple, la modification du document ou son résumé) nécessite l'autorisation préalable et expresse des auteurs ou de leurs ayants droit.

Towards a distributed implementation of a data extraction system for monitoring the processing of video clips

Vasbinder Thomas

Adviser: Prof. Pierre Wolper

Master in computer science and engineering

Academic year 2015-2016

The objective of this work was to review the data handling of an application monitoring the processing of video clips and to remove any barrier preventing the concurrent execution of multiple instances of the service, while maintaining or improving the overall performance.

The main obstacle to concurrent execution was that the caching infrastructure didn't provide synchronisation between the cache and the database. We showed that the database could perform queries in memory like the in-memory cache at the server. The database executed most of the queries in a more efficient and predictable way than the cache, thanks to its query optimiser.

Based on those results, we decided to remove the cache and perform queries directly on the database. This removal simplified and shortened the code of the application. However, there was no mechanism of translation of the query contained in the URI to the database. We thus used OData, a tool providing a query language for RESTful APIs, that can translate automatically query parameters from the URI to SQL queries. Yet, OData came with some performance flaws. To restore performances, we used indexes to speed up queries execution and we restricted query parameters combinations.

This work resulted in a solution allowing the concurrent execution of multiple instances of the service. Moreover, queries are in most cases faster and thus the service can now support more clients.

So, this work showed that the cache was not better than the database when executing the queries and could be removed.