

# RAPPORT DE STAGE

MASTER 2 MIAGE parcours IA2

TOWARDS A HOLISTIC BENCHMARKING PLATFORM  
FOR SEMANTIC WEB DATA STORAGE SYSTEMS  
SAUNDERS PIERRE

15/04/2021 – 02/09/2021

**Tuteur du stage en entreprise** : Monsieur Damien GRAUX

**Tuteur académique** : Madame Elena CABRIO

**Etablissement / Formation** : Université Côte d'Azur / DS4H

**Entreprise d'accueil** : Inria

# Sommaire

<b>1 Remerciements</b>	<b>3</b>
<b>2 Introduction</b>	<b>3</b>
<b>3 Partie 1 - L'entreprise et son secteur d'activité</b>	<b>3</b>
3.1 Présentation de l'entreprise	3
3.1.1 À propos du centre	3
3.1.2 Matrice SWOT	4
3.1.3 Organigramme	5
3.2 Analyse externe de l'entreprise	5
3.3 Contexte de stage	5
<b>4 Partie 2 - Mon Stage</b>	<b>6</b>
4.1 Objectif du stage	6
4.2 Les missions	6
4.2.1 Dockerisation de moteurs	7
4.2.2 Dockerisation de benchmarks	10
4.2.3 Dockerisation d'outils	10
4.2.4 Plateforme de benchmark	10
4.3 Le bilan	11
4.3.1 Résultats obtenus	11
4.3.2 Difficultés rencontrées et solutions apportées	11
4.3.3 Enseignements/apports du stage	12
<b>5 Conclusions</b>	<b>12</b>
<b>6 Annexes / Bibliographie</b>	<b>12</b>

## 1 Remerciements

Je souhaite remercier dans un premier temps M. Damien GRAUX chercheur titulaire à l'Inria et également mon tuteur de stage, pour sa confiance et les connaissances qu'il a su partager avec moi. Je le remercie aussi pour sa disponibilité et la qualité de son encadrement au sein du laboratoire. Je tiens aussi à remercier toute l'équipe Wimmics pour son accueil.

## 2 Introduction

Cela fait plusieurs mois que j'étais curieux de découvrir le monde de la recherche, j'ai donc naturellement voulu faire un stage en laboratoire et je me suis tourné vers les laboratoires Inria [1] et I3S [2]. Ce sont des laboratoires où certains de mes enseignants travaillent et ces laboratoires ont l'avantage d'être proche géographiquement. C'est alors que M. Damien GRAUX du laboratoire Inria m'a contacté, ayant vu mon portfolio, il était intéressé par mon profil et nous avons donc parlé de son sujet de stage. J'ai immédiatement accroché tant les technologies utilisées m'étaient inconnues mais intrigantes. Ce rapport vise à présenter le laboratoire Inria et à résumer mon travail durant ce stage.

## 3 Partie 1 - L'entreprise et son secteur d'activité

### 3.1 Présentation de l'entreprise

Inria [1] est l'institut national de recherche dédié aux sciences et aux technologies du numérique. Il emploie 2 600 personnes. Ses 200 équipes-projets agiles, en général communes avec des partenaires académiques, impliquent plus de 3 500 scientifiques pour relever les défis du numérique, souvent à l'interface d'autres disciplines. L'institut fait appel à de nombreux talents dans plus d'une quarantaine de métiers différents. 900 personnels d'appui à la recherche et à l'innovation contribuent à faire émerger et grandir des projets scientifiques ou entrepreneuriaux qui impactent le monde. Inria [1] travaille avec de nombreuses entreprises et a accompagné la création de plus de 180 start-ups. L'institut s'efforce ainsi de répondre aux enjeux de la transformation numérique de la science, de la société et de l'économie.

#### 3.1.1 À propos du centre

Le centre Inria Sophia-Antipolis – Méditerranée [3] compte 34 équipes de recherche ainsi que 8 départements de soutien. Le personnel du centre (environ 500 personnes dont 320 employés Inria [1]) est composé de scientifiques de différentes nationalités (250 étrangers de 50 nationalités), d'ingénieurs, de techniciens et de personnel administratif. 1/3 du personnel est fonctionnaire, les autres sont des agents contractuels. La majorité des équipes de recherche du centre sont localisées à Sophia-Antipolis et à Nice dans les

Alpes-Maritimes. Quatre équipes sont hébergées à Bologne en Italie et à Athènes. Le centre est membre fondateur d'Université Côte d'Azur et partenaire de l'I-site MUSE soutenu par l'Université de Montpellier.

### 3.1.2 Matrice SWOT

<b>Interne</b>	
<p><b><u>Forces (i.e. Strengths)</u></b></p> <ul style="list-style-type: none"> <li>• Des employés talentueux et dévoués</li> <li>• Un haut niveau d'engagement au service des étudiants</li> <li>• Une structure de gouvernance efficace et bien définie</li> <li>• Excellente réputation</li> <li>• Un fort sens de la famille</li> </ul>	<p><b><u>Opportunités (i.e. Opportunities)</u></b></p> <ul style="list-style-type: none"> <li>• Sensibilisation efficace des marchés existants et nouveaux</li> <li>• Planification et coordination d'événements</li> <li>• Participe à des programmes nationaux pour l'innovation</li> </ul>
<b>Externe</b>	
<p><b><u>Faiblesses (i.e. Weaknesses)</u></b></p> <ul style="list-style-type: none"> <li>• Structure et processus d'octroi de subventions</li> <li>• Les installations sont limitées aux heures de pointe</li> <li>• La taille de la base d'employés rend impossible la tenue d'une réunion de tout le consortium sur le site, ce qui complique la communication.</li> <li>• L'intégration de la budgétisation et de la planification évolue</li> </ul>	<p><b><u>Menaces (i.e. Threats)</u></b></p> <ul style="list-style-type: none"> <li>• Le financement de l'État est limité et instable</li> <li>• D'autres institutions semblent avoir un budget marketing illimité</li> <li>• Législation(s) ayant un impact négatif sur nous au niveau de l'État ou du gouvernement fédéral.</li> <li>• Se concentrer sur le succès plutôt que sur l'accès sera un défi (et pourrait être financièrement difficile).</li> </ul>

Table 1: Matrice SWOT

### 3.1.3 Organigramme

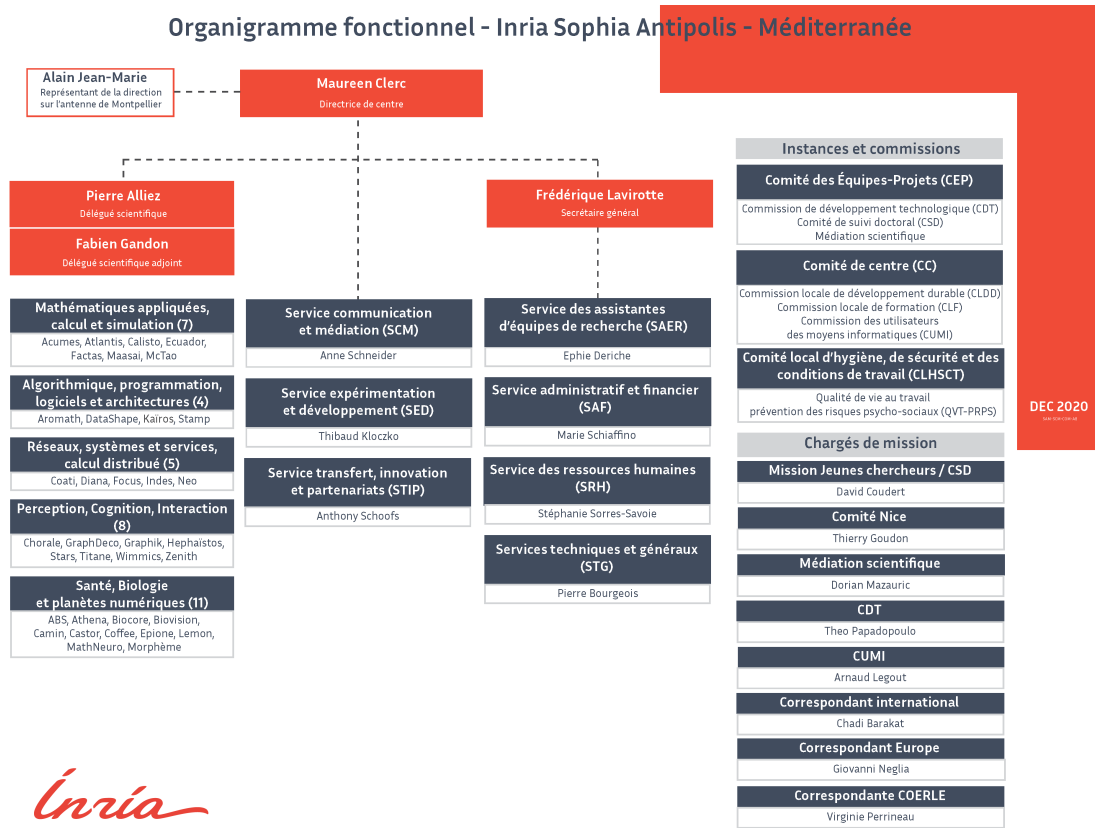


Figure 1: Organigramme d'Inria Sophia Antipolis - Méditerranée (Source : intranet Inria)

## 3.2 Analyse externe de l'entreprise

Inria est un laboratoire de recherche au financement public et est par conséquent inéligible à l'analyse des 5 forces de PORTER ainsi que le biais de la matrice PESTEL.

## 3.3 Contexte de stage

SPARQL [4] est le langage standard pour les requêtes de données sémantique RDF [5]. Il existe une variété de systèmes d'évaluation de requêtes SPARQL [4] (i.e. moteur RDF [5]) mettant en œuvre différentes architectures [6]. Les différences d'architectures couplées à des optimisations spécifiques, par exemple pour le prétraitement et l'indexation, rendent ces systèmes incomparables d'un point de vue purement théorique. Il en résulte que de nombreuses implémentations résolvent le problème de l'évaluation des requêtes SPARQL [4] tout en présentant des comportements très différents, toutes n'étant pas adaptées à aucun contexte [7]. À ce jour, une pléthore de benchmark (i.e. un test standardisé qui sert de base à

l'évaluation ou à la comparaison) a été conçue afin de guider les utilisateurs dans le choix de systèmes les mieux adaptés à leurs besoins [8]. Malheureusement, la communauté manque actuellement d'un endroit où les systèmes pourraient être ouvertement et équitablement comparés les uns aux autres en utilisant ces benchmarks publics. Ce stage vise principalement à établir une telle plateforme.

## 4 Partie 2 - Mon Stage

### 4.1 Objectif du stage

En pratique, le sujet du stage est double. Premièrement, le projet consiste à concevoir un ensemble de routines pour déployer de manière transparente des solutions de stockage RDF [5] et tester leurs capacités d'évaluation SPARQL [4], permettant ainsi une comparaison équitable de leurs performances sous différentes configurations. Deuxièmement, les résultats obtenus à partir de l'environnement de test peuvent être utilisés pour extraire de nouvelles conclusions en considérant par exemple des formes spécifiques de requêtes SPARQL [4] :

- Sélectionner (à partir de l'état de l'art) des évaluateurs/points finaux et des benchmarks potentiels à inclure dans la plateforme.
- Développer les scripts et les fichiers permettant un déploiement rapide de ces solutions.
- Exécuter les benchmarks et établir les premières analyses sur la base de l'ensemble des métriques enregistrées.
- Structurer l'ensemble des résultats de ces performances.
- En tirer des enseignements afin que par exemple, de prédire quel type d'évaluateur pourrait convenir à une requête spécifique.

En outre, les résultats pourraient être partagés en ligne, offrant ainsi à la communauté une ressource utile. Cette interface pourrait également fournir aux utilisateurs un formulaire (ou un ensemble de curseurs) leur permettant de décrire leur cas d'utilisation afin de se voir recommander une éventuelle solution appropriée.

### 4.2 Les missions

Pour pouvoir comparer les différents moteurs RDF [6], il est naturel de les avoir à disposition sur une machine locale où l'on peut mesurer les différentes métriques qui nous intéressent. Et pour cela, nous avons plusieurs options, la première, la plus naturelle, consiste tout simplement à télécharger et à installer directement sur la dite machine, mais cela implique de pouvoir faire coexister les différents moteurs sur un même système d'exploitation avec les dépendances de chacun sans problème d'incompatibilité. Plus le nombre de moteurs augmentera, plus le risque que cela n'arrive est probable. Une autre solution pour remédier à cela est d'utiliser une machine virtuelle pour chaque moteur, mais cela rend la mesure des

métriques très ardue, et la virtualisation des systèmes d'exploitation interne risque de biaiser les résultats. Nous allons donc utiliser Docker [9] qui permet de garder cette idée de virtualisation, mais exécuter en plus bas niveau, au niveau du noyau Linux. Cette technique s'appelle la conteneurisation, et cela permet d'avoir le meilleur des deux mondes cités précédemment.

#### 4.2.1 Dockerisation de moteurs

##### 4store

4store [10] est un triple store natif qui propose sa propre logique de stockage. Il s'appuie sur le partage de données de manière égale entre chaque machine du cluster. Les données sont indexées et écrites sur le disque. Au moment de la requête, le conducteur interprète la requête et dégèle la résolution à tous les travailleurs, puis agrège les résultats intermédiaires renvoyés par le travailleur pour produire le résultat final [11].

##### RDF3X

RDF-3X [12] (i.e. RDF-3X engine) est une implémentation de SPARQL [4] qui atteint d'excellentes performances en poursuivant une architecture de style RISC (i.e. Reduced instruction set computer) avec une architecture rationalisée et des structures de données et opérations puristes soigneusement conçues. Les points saillants de RDF-3X [12] sont : 1) une solution générique pour le stockage et l'indexation des triplets RDF [5] qui élimine complètement le besoin de réglage de la conception physique, 2) un processeur de requête puissant mais simple qui exploite les jointures de fusion rapides dans la plus grande mesure possible, et 3) un optimiseur de requête pour choisir les ordres de jointures optimaux en utilisant un modèle de coût basé sur des synopsis statistiques pour les chemins de jointure entiers [13].

##### RDF4J

RDF4J [14] (i.e. Eclipse RDF4J) est un puissant framework Java pour le traitement et la manipulation de données RDF [5]. Cela comprend la création, l'analyse syntaxique, le stockage évolutif, le raisonnement et l'interrogation avec RDF [5] et Linked Data. Il offre une API facile à utiliser qui peut être connectée à toutes les principales solutions de bases de données RDF [5]. Il vous permet de vous connecter avec des points de terminaison SPARQL [4] (i.e. endpoint) et de créer des applications qui tirent parti de la puissance des données liées et du Web sémantique.

##### Apache Jena

Jena [15] (i.e. Apache Jena) est un framework Java pour la création d'applications Web sémantiques. Il fournit de nombreuses bibliothèques Java pour aider les développeurs à développer du code qui gère RDF [5], RDFS [16], RDFa [17], OWL [18] et SPARQL [4] conformément aux recommandations publiées par le W3C. Jena comprend un moteur d'inférence à base de règles [19] pour effectuer un raisonnement basé

sur les ontologies OWL [18] et RDFS [16], et une variété de stratégies de stockage pour stocker les triplets RDF [5] en mémoire ou sur disque. Jena [15] a été développé à l'origine par des chercheurs de HP Labs [20], à partir de Bristol, au Royaume-Uni, en 2000. Jena [15] a toujours été un projet open-source, et a été largement utilisé dans une grande variété d'applications et de démonstrateurs du web sémantique. En 2009, HP a décidé de réorienter ses activités de développement et de ne plus soutenir directement le développement de Jena [15], tout en continuant à soutenir les objectifs du projet. L'équipe du projet a réussi à faire adopter Jena [15] par l'Apache Software Foundation [21] en novembre 2010 (voir le résultat du vote). Jena [15] est entré en incubation avec l'Apache Software Foundation [21] en novembre 2010, et est devenu un projet de haut niveau en avril 2012.

### **Virtuoso**

Virtuoso [22] (i.e. OpenLink Software Virtuoso) est une plateforme moderne construite sur des normes ouvertes existantes qui exploite la puissance des hyperliens (fonctionnant comme des super clés) pour éliminer les silos de données qui entravent les capacités des utilisateurs et des entreprises. Par exemple, le noyau SQL et SPARQL [4] de Virtuoso alimente de nombreuses initiatives de graphe de connaissances d'entreprise, tout comme il alimente DBpedia [23] et la majorité des nœuds du Linked Open Data Cloud [24] – le plus grand graphe de connaissances accessible au public au monde.

### **Apache Jena Fuseki**

Apache Jena Fuseki [25] est un serveur SPARQL [4]. Il peut fonctionner comme un service du système d'exploitation, comme une application web Java (fichier WAR), et comme un serveur autonome. Fuseki se présente sous deux formes, un système unique "webapp" combiné avec une interface utilisateur pour l'administration et l'interrogation, et comme "main", un serveur adapté pour fonctionner dans le cadre d'un déploiement plus large. Les deux formes utilisent le même moteur de protocole de base et le même format de fichier de configuration. Fuseki fournit les protocoles SPARQL [4] 1.1 pour les requêtes et les mises à jour, ainsi que le protocole SPARQL Graph Store [26]. Fuseki est étroitement intégré à TDB [27] pour fournir une couche de stockage persistance transactionnelle et robuste, et incorpore la requête textuelle Jena.

### **Corese**

Corese [28] (i.e. Conceptual Resource Search Engine) est une plateforme logicielle pour le Web sémantique des données liées créée par l'équipe Wimmics [29] à l'Inria [1] disponible en open-source [30]. Il fournit RDF [5], RDFS [16], SPARQL Query & Update [31], SPARQL Inference Rules [19], SPARQL Template Pretty Printing Language [32] et plusieurs extensions. Corese [28] est la suite du projet KGRAM [33] (i.e. Knowledge Graph Abstract Machine).



## SPARQLGX

SPARQLGX [34] est l'implémentation par l'équipe Tyrex [35] d'un datastore RDF [5] distribué basé sur Apache Spark [36]. SPARQLGX [34] est conçu pour exploiter les infrastructures Hadoop [37] existantes pour l'évaluation de requêtes SPARQL [4]. SPARQLGX [34] repose sur une traduction des requêtes SPARQL en code exécutable Spark [36] qui adopte des stratégies d'évaluation en fonction (1) de la méthode de stockage utilisée et (2) des statistiques sur les données. Avec un design simple, SPARQLGX [34] représente déjà une alternative intéressante dans plusieurs scénarios.

## CumulusRDF

CumulusRDF [38] est un store RDF [5] open source [39], conçu pour les applications à forte intensité d'écriture. Il est construit au-dessus d'Apache Cassandra [40] et supporte entièrement le stockage hautement évolutif de triples et quads RDF [5]. CumulusRDF [38] fournit différentes interfaces pour effectuer des opérations de manipulation de données (chargement, vidage, suppression et interrogation). Le langage d'interrogation supporté par CumulusRDF [38] est SPARQL [4]. CumulusRDF [38] est capable d'exécuter des requêtes composées de Basic Graph Patterns (i.e. BGP [41]), c'est-à-dire que la clause WHERE ne contient que des séquences de motifs triples. CumulusRDF [38] peut être déployé comme une application Web, en utilisant un serveur Apache Tomcat [42]. Il offre un point de terminaison SPARQL [4], qui peut traiter les requêtes spécifiées en utilisant le protocole SPARQL [4]. CumulusRDF [38] dispose également d'une interface graphique Web, qui fournit des rapports de synthèse sur les statistiques de données et les performances du triple store. Grâce à l'interface graphique Web, les utilisateurs peuvent télécharger des données et interroger le triple store.

## Mulgara

Mulgara [43] [44] met en œuvre de nombreux concepts du Web sémantique du World Wide Web Consortium (i.e. W3C [45]). Les bases de données Mulgara [43] contiennent des métadonnées sous la forme de courtes déclarations sujet-prédicat-objet, un peu comme la norme RDF [5] du W3C [45]. En outre, les métadonnées peuvent être importées dans TKS sous forme de RDF [5]. Grâce aux commandes iTQLM (i.e. Interactive Tucana Query Language), vous pouvez interroger les bases de données Mulgara [43] et recevoir les résultats correspondant à la requête. iTQL est similaire au langage SQL (i.e. Structured Query Language) utilisé pour interroger les bases de données relationnelles, avec quelques différences significatives dues à la manière dont les données sont stockées dans Mulgara [43]. Comme les bases de données relationnelles, Mulgara [43] peut être utilisé comme un référentiel de données sous-jacent pour les applications logicielles.

#### 4.2.2 Dockerisation de benchmarks

##### LUBM

Le “Lehigh University BenchMark” [46] (i.e. LUBM) a été développé pour faciliter l’évaluation des référentiels du Web sémantique de manière standard et systématique [47]. Le benchmark est destiné à évaluer la performance de ces référentiels en ce qui concerne les requêtes extensionnelles sur un grand ensemble de données qui s’engage dans une seule ontologie réaliste [48]. Il s’agit d’une ontologie de domaine universitaire, de données synthétiques personnalisables et répétables, un ensemble de requêtes de test et plusieurs métriques de performances.

##### SP2Bench

SP2Bench [49] est un benchmark de performance pour le langage de requête SPARQL [4]. Le benchmark comprend à la fois un générateur de données permettant de générer de manière arbitraire des ensembles de données de type DBLP [50] et des ensembles de requêtes SPARQL [4] [2] de référence, qui couvrent une variété de configurations d’opérateur SPARQL [4] et de modèles d’accès RDF [5] typiques [50].

##### WatDiv

WatDiv [51] mesure la performance d’un système de gestion de données RDF [5] sur un large spectre de requêtes SPARQL [4] avec des caractéristiques structurales et des classes de sélectivité variables [52]. WatDiv [51] se compose de deux éléments : le générateur de données et le générateur de requêtes (et de modèles). Le générateur de données WatDiv [51] permet aux utilisateurs de définir leur propre jeu de données par le biais d’une description du jeu de données.

#### 4.2.3 Dockerisation d’outils

##### RDF2RDF

Le but de l’outil RDF2RDF [53] est de fournir un moyen facile de convertir une sérialisation RDF [5] en une autre. Il utilise le jar RDF2RDF [53] développé par Enrio Manick il y a plus de dix ans. Cette reprise vise à fournir une dockerisation du jar car les versions actuelles de Java ne sont pas capables de l’exécuter et le code source n’est plus disponible.

#### 4.2.4 Plateforme de benchmark

Maintenant que nous avons dockerisé les moteurs et benchmarks et harmonisé leur utilisation. Il était temps de concevoir les balbutiements d’une plateforme de benchmark. Chaque moteur et benchmark étant chacun dans un répertoire GitHub [54] respectif, nous avons donc créé un script UNIX [55] qui clone les répertoires, construit les images Docker [9] et exécute les benchmarks sur chacun des moteurs tout en mesurant le temps de chargement du set de données, le temps d’évaluation d’une requête, l’écriture/lecture disque, etc.

Toutes les mesures sont disponibles dans un fichier CSV et les logs supplémentaires dans un sous-dossier. Il est également possible de tester différentes contraintes comme la limite de mémoire vive (500 MB, 1 GB, 2 GB, etc.), le nombre de cœurs CPU alloués (1 à 4) ou encore la mise en panne artificielle d'un nœud (pour le moteur travaillant sur plusieurs nœuds). Cela permettra de mesurer d'autres métriques comme la résilience.

## 4.3 Le bilan

### 4.3.1 Résultats obtenus

J'ai finalement dockerisé et normalisé 10 moteurs RDF [5] et SPARQL endpoints [26]. Par manque de temps, je n'ai pas pu tronquer les images de SPARQLGX [34], CumulusRDF [38] [39] et Mulgara [43] [44].

Nom de l'image	Taille (MB)
<b>Moteurs</b>	
4store [11] [10]	167
RDF3X [12] [13]	96
RDF4J [14]	73
Apache Jena [15]	79
Virtuoso [22]	138
Apache Jena Fuseki [25]	147
Corese [28] [30]	216
SPARQLGX [34]	3380
CumulusRDF [38] [39]	869
Mulgara [43] [44]	186
<b>Benchmarks</b>	
LUBM [47] [46]	38
SP2Bench [50] [49]	76
WatDiv [51]	99
<b>Outils</b>	
RDF2RDF [53]	105

Figure 2: Taille des images Docker [9]

Même s'il reste d'avantages de moteurs RDF [5] et SPARQL endpoints [26] à intégrer, cela donnent déjà une bonne base de comparaison pour le futur du projet.

### 4.3.2 Difficultés rencontrées et solutions apportées

Les principales difficultés rencontrées ont été les mêmes pour chaque moteur et benchmark, il fallait d'abord réussir à dockeriser l'application, ce qui peut être compliqué selon les différentes manières d'installation (C, C++, Objective C, Java, Maven etc...), ensuite, il faut tronquer l'image pour ne garder que le minimum et ainsi réduire la taille de l'image, cela s'est fait majoritairement avec des images Alpine [56], un système d'exploitation ultra léger, mais qui a l'inconvénient d'utiliser les bibliothèques musl [57] et non les bibliothèques glibc [58]. Et enfin normaliser son utilisation, cette dernière étape est certainement la plus compliquée car, malgré

les standards du web sémantique [4] [5] [26] [16] [31] [41], chaque application à sa manière propre de faire les choses. Malgré tout, la manière de normaliser les images est restée inchangée du début à la fin du stage, ce qui permet de facilement évaluer chaque moteur avec chaque benchmark.

### 4.3.3 Enseignements/apports du stage

Ce stage m'a permis d'apprendre en détail l'environnement Docker [9], comment dockeriser une application, minimiser l'image résultante et également d'apprendre à utiliser en détail les différents triples stores et SPARQL endpoints [26] disponibles. J'ai également appris à utiliser LaTeX [59] pour la rédaction de ce rapport.

## 5 Conclusions

Pour conclure, les objectifs du stage étant de dockeriser et normaliser les différents moteurs RDF [5] et SPARQL endpoints [26] dans une plateforme de benchmark sont remplis. Ce stage m'a beaucoup appris sur le monde du web sémantique et m'a permis de poser une modeste pierre sur l'édifice du monde de la recherche. Ce stage a également confirmé mon souhait de vouloir enchaîner sur une thèse et à terme devenir chercheur.

## 6 Annexes / Bibliographie

- [1] Inria, *Institut National de Recherche en Informatique et en Automatique*. [Online]. Available: <https://www.inria.fr/fr>
- [2] I3S, *I3S official website*. [Online]. Available: <https://www.i3s.unice.fr>
- [3] Inria, *Inria Sophia Antipolis official website*. [Online]. Available: <https://www.inria.fr/en/centre-inria-sophia-antipolis-mediterranee>
- [4] W3C, *SPARQL Standard*. [Online]. Available: <https://www.w3.org/TR/sparql11-overview>
- [5] —, *RDF Primer*. [Online]. Available: <https://www.w3.org/TR/rdf11-primer>
- [6] *RDF Data Storage and Query Processing Schemes : A Survey*. [Online]. Available: <https://dl.acm.org/doi/pdf/10.1145/3177850>
- [7] D. Graux, L. Jachiet, P. Genevès, and N. Layaïda, "A Multi-Criteria Experimental Ranking of Distributed SPARQL Evaluators," in *Big Data 2018 - IEEE International Conference on Big Data*. Seattle, United States: IEEE, Dec. 2018, pp. 1–10. [Online]. Available: <https://hal.inria.fr/hal-01381781>

- [8] M. Saleem, G. Szárnyas, F. Conrads, S. A. C. Bukhari, Q. Mehmood, and A.-C. Ngonga Ngomo, “How representative is a sparql benchmark? an analysis of rdf triplestore benchmarks,” in *The World Wide Web Conference*, ser. WWW '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1623–1633. [Online]. Available: <https://doi.org/10.1145/3308558.3313556>
- [9] *Docker official website*. [Online]. Available: <https://www.docker.com>
- [10] *4store's GitHub repository*. [Online]. Available: <https://github.com/SemanticWebBenchmark/4store>
- [11] S. Harris, N. Lamb, and N. Shadbolt, “4store: The design and implementation of a clustered rdf store,” in *IN: SCALABLE SEMANTIC WEB KNOWLEDGE BASE SYSTEMS - SSWS2009*, 2009, pp. 94–109. [Online]. Available: <https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.368.5588>
- [12] *RDF3X's GitHub repository*. [Online]. Available: <https://gitlab.db.in.tum.de/dbtools/rdf3x>
- [13] T. Neumann and G. Weikum, “Rdf-3x: A risc-style engine for rdf,” *Proc. VLDB Endow.*, vol. 1, no. 1, p. 647–659, Aug. 2008. [Online]. Available: <https://doi.org/10.14778/1453856.1453927>
- [14] *RDF4J's official website*. [Online]. Available: <https://rdf4j.org>
- [15] Apache, *Apache Jena's official website*. [Online]. Available: <https://jena.apache.org/index.html>
- [16] W3C, *RDF Schema*. [Online]. Available: <https://www.w3.org/TR/rdf-schema>
- [17] —, *RDFa Standard*. [Online]. Available: <https://www.w3.org/2001/sw/wiki/RDFa>
- [18] —, *OWL Standard*. [Online]. Available: <https://www.w3.org/TR/owl-features>
- [19] W. M. Submission, *SPARQL Inferencing Notation*. [Online]. Available: <https://spinrdf.org>
- [20] HP, *HP Labs official website*. [Online]. Available: <https://www.hp.com/us-en/hp-labs/index.html>
- [21] Apache, *Apache's official website*. [Online]. Available: <https://apache.org>
- [22] *Virtuoso's official website*. [Online]. Available: <https://virtuoso.openlinksw.com>
- [23] *Dbpedia's official website*. [Online]. Available: <https://www.dbpedia.org>
- [24] *The Linked Open Data Cloud's official website*. [Online]. Available: <https://www.lod-cloud.net>
- [25] Apache, *Apache Jena Fueski's official website*. [Online]. Available: <https://jena.apache.org/documentation/fuseki2>
- [26] *SPARQL 1.1 Graph Store HTTP Protocol Standard*. [Online]. Available: <https://www.w3.org/TR/sparql11-http-rdf-update>
- [27] Apache, *TDB2's official website*. [Online]. Available: <https://jena.apache.org/documentation/tdb2>

- [28] Inria, *Corese's official website*. [Online]. Available: <https://project.inria.fr/corese>
- [29] Wimmics, *Wimmics – Bridging social semantics and formal semantics on the web*. [Online]. Available: <https://team.inria.fr/wimmics>
- [30] Inria, *Corese's GitHub repository*. [Online]. Available: <https://github.com/Wimmics/corese>
- [31] W3C, *SPARQL Query Language for RDF*. [Online]. Available: <https://www.w3.org/TR/rdf-sparql-query>
- [32] O. Corby and C. Faron Zucker, “SPARQL Template : un langage de Pretty Printing pour RDF,” in *IC - 25èmes Journées francophones d'Ingénierie des Connaissances*, C. Faron-Zucker, Ed., Clermont-Ferrand, France, May 2014, pp. 213–224, session 4 : Web sémantique. [Online]. Available: <https://hal.inria.fr/hal-01015267>
- [33] Inria, *KGRAM's GitHub repository*. [Online]. Available: <https://gforge.inria.fr/projects/kgram/>
- [34] Tyrex, *SPRAQLGX's official website*. [Online]. Available: <https://tyrex.inria.fr/sparqlgx/home.html>
- [35] —, *Tyrex's official website*. [Online]. Available: <https://tyrex.inria.fr/>
- [36] Apache, *Apache SPARK's official website*. [Online]. Available: <https://spark.apache.org>
- [37] —, *Apache Hadoop's official website*. [Online]. Available: <https://hadoop.apache.org>
- [38] K. I. für Technologie, *CumulusRDF's official website*. [Online]. Available: <https://www.aifb.kit.edu/web/CumulusRDF/en>
- [39] —, *CumulusRDF's GitHub repository*. [Online]. Available: <https://github.com/cumulusrdf/cumulusrdf>
- [40] Apache, *Apache Cassandra's official website*. [Online]. Available: [https://cassandra.apache.org/\\_/index.html](https://cassandra.apache.org/_/index.html)
- [41] W3C, *SPARQL - Graph Patterns*. [Online]. Available: <https://www.w3.org/2004/Talks/17Dec-sparql/QueryLang1/all.html>
- [42] Apache, *Apache Tomcat's official website*. [Online]. Available: <https://tomcat.apache.org>
- [43] Mulgara, *Mulgara's official website*. [Online]. Available: <http://docs.mulgara.org/system/index.html>
- [44] —, *Mulgara's GitHub repository*. [Online]. Available: <https://github.com/quoll/mulgara>
- [45] W3C, *W3C's official website*. [Online]. Available: <https://www.w3.org>
- [46] *LUBM's official website*. [Online]. Available: <http://sc.cse.lehigh.edu/projects/lubm>

- [47] Y. Guo, Z. Pan, and J. Heflin, “Lubm: A benchmark for owl knowledge base systems,” *Journal of Web Semantics*, vol. 3, no. 2, pp. 158–182, 2005, selected Papers from the International Semantic Web Conference, 2004. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1570826805000132>
- [48] —, “An evaluation of knowledge base systems for large owl datasets,” in *The Semantic Web – ISWC 2004*, S. A. McIlraith, D. Plexousakis, and F. van Harmelen, Eds., vol. 3298. Berlin, Heidelberg: Springer Berlin Heidelberg, 11 2004, pp. 274–288.
- [49] *SP2Bench’s official website*. [Online]. Available: <http://dbis.informatik.uni-freiburg.de/index.php?project=SP2B>
- [50] M. Schmidt, T. Hornung, G. Lausen, and C. Pinkel, “Sp2bench: A SPARQL performance benchmark,” *CoRR*, vol. abs/0806.4627, 2008. [Online]. Available: <http://arxiv.org/abs/0806.4627>
- [51] *WatDiv’s official website*. [Online]. Available: <https://dsg.uwaterloo.ca/watdiv>
- [52] S. Duan, A. Kementsietsidis, K. Srinivas, and O. Udrea, “Apples and oranges: A comparison of rdf benchmarks and real rdf datasets,” in *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD ’11. New York, NY, USA: Association for Computing Machinery, 2011, p. 145–156. [Online]. Available: <https://doi.org/10.1145/1989323.1989340>
- [53] *RDF2RDF’s official website*. [Online]. Available: <http://www.l3s.de/~minack/rdf2rdf>
- [54] D. Graux and P. Saunders, *Team’s GitHub*, 2021. [Online]. Available: <https://github.com/SemanticWebBenchmark>
- [55] *The UNIX® Standard*. [Online]. Available: <https://www.opengroup.org/membership/forums/platform/unix>
- [56] Alpine, *Alpine’s official website*. [Online]. Available: <https://alpinelinux.org/about>
- [57] musl, *musl’s official website*. [Online]. Available: <https://musl.libc.org>
- [58] GNU, *GNU C library’s official website*. [Online]. Available: <https://www.gnu.org/software/libc>
- [59] LaTeX, *LaTeX’s official website*. [Online]. Available: <https://www.latex-project.org>