

Négociation de contenu sémantique pour l'échange de connaissances entre systèmes hétérogènes

Y. Taghzouti¹, A. Zimmermann¹, M. Lefrançois¹

¹ Mines Saint-Étienne, Univ Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS, F - 42023 Saint-Étienne France

yousouf.taghzouti@emse.fr, antoine.zimmermann@emse.fr, maxime.lefrancois@emse.fr

Résumé

*Les ressources sur le Web sont identifiées par des identifi-
cateurs uniformes de ressources (URI). Chaque ressource
peut avoir plusieurs représentations la décrivant, que nous
appelons variantes. Un utilisateur (client) souhaitant une
représentation particulière de cette ressource fait une re-
quête à son URI avec un ensemble de contraintes. La
négociation de contenu (NC) est le processus par lequel
un serveur fait correspondre les préférences du client à
l'ensemble des variantes de la ressource. Pendant long-
temps, le type de média et la langue étaient les principales
contraintes qui différenciaient les variantes. Plus tard, des
variantes ont été créées pour représenter différentes ver-
sions évolutives dans le temps de la ressource avec le même
type de média et la même langue. De plus, aujourd'hui,
avec l'utilisation des technologies du web sémantique, no-
tamment RDF, une ressource peut être décrite à l'aide de
plusieurs vocabulaires et ontologies et en adhérant à dif-
férents profils de langage d'ontologie web. Une négocia-
tion plus fine de contenu est donc nécessaire, et la réponse
du serveur doit être flexible, notamment pour répondre à
une requête si aucune représentation ne valide parfaitement
toutes les contraintes. Dans ce travail, nous présentons les
défis associés à la résolution du problème de la négocia-
tion fine de contenu dans un environnement web hétérogène.
Nous présentons nos questions de recherche pour résoudre
ce problème et les hypothèses proposées. Enfin, nous décri-
vons notre méthodologie, y compris un plan d'évaluation,
et fournissons les résultats préliminaires obtenus et la di-
rection que nous envisageons de prendre.*

Mots-clés

Négociation de contenu, web sémantique, contrainte.

Abstract

*Resources on the Web are identified by uniform resource
identifiers (URIs). Each resource can have several repre-
sentations describing it, which we call variants. A user
(client) wanting a particular representation of that re-
source makes a request to its URI along with a set of
constraints. Content negotiation is the process by which
a server matches the client's preferences to the set of va-
riants of the resource. For a long time, media type and lan-*

*guage were the primary constraints that differentiated the
variants. Later, variants were created to represent different
evolving versions over time of the resource with the same
media type and language. Furthermore, today, with the use
of semantic web technologies, especially RDF, a resource
can be described using multiple vocabularies and ontolo-
gies and adhering to different web ontology language pro-
files. A finer negotiation of the content is therefore neces-
sary, and the server response must be flexible, especially to
answer a request if no representation perfectly validates all
the constraints. In this work, we present the challenges as-
sociated with solving the fine-grained content negotiation
problem in a heterogeneous web environment. We present
our research questions to solve this problem and the propo-
sed hypotheses. Finally, we describe our methodology, in-
cluding an evaluation plan, and provide preliminary results
obtained and the direction we plan to take.*

Keywords

Content negotiation, semantic web, constraint.

1 Introduction

Ouvert, distribué, accessible et hétérogène sont quelques-unes des caractéristiques fondamentales du Web [4]. Bien que le fait que n'importe qui puisse accéder au Web de n'importe où dans le monde ait grandement contribué à son développement et à son enrichissement grâce à son ouverture, cela a eu l'effet indésirable d'avoir une abondance de ressources Web et des difficultés à fournir le meilleur contenu pour chaque client ; un exemple simple est celui de deux personnes parlant des langues différentes accédant à la même ressource. Dans ce cas, le serveur avec la ressource devrait être capable de fournir à chaque client une version compréhensible. Pour remédier à cela, une solution a été imaginée dès le départ, avec une couche de négociation entre le client et le serveur [3]. Elle est décrite dans le document Architecture of the World Wide Web comme l'un des composants essentiels de la conception du Web [14, section 3.2].

La négociation, en tant que concept, est une communication aller-retour destinée à atteindre un accord lorsque deux ou plusieurs parties ont des intérêts communs et d'autres opposés [9, p. 1]. Appliqué au Web, elle devient alors le

mécanisme permettant de sélectionner la représentation appropriée lors du traitement d'une requête. Dans HTTP, on peut exprimer et transmettre des contraintes appelées préférences dans [8, section 5.3]. Et avec cela, en plus de trouver et de transmettre des informations, il est possible de sélectionner des formats et des langages plus spécifiques.

Avec l'ère du mobile, un nouveau défi est apparu, le contenu déjà disponible était conçu pour s'adapter aux écrans d'ordinateurs et non aux téléphones. Une fois encore, il a fallu négocier le contenu pour savoir ce qui convenait à ces appareils en fonction de leurs caractéristiques [15].

Les ressources sur le Web étaient principalement destinées aux humains, pas aux machines. L'absence d'un contenu sémantiquement compréhensible a empêché leur exploitation complète, mais cela a changé avec le développement de langages web sémantiques pour décrire le contenu du Web et fournir un moyen pour les machines de le comprendre. Pour ce faire, on a d'abord utilisé le Resource Description Framework (RDF) [12]. Puis, en utilisant le Web Ontology Language (OWL) et une variété de vocabulaires [11]. Néanmoins, cette diversité a révélé le besoin d'une négociation fine qui va au-delà du simple format ou langage tel qu'il était.

Dans ce travail de thèse, nous nous intéressons à la manière de rendre la négociation de contenu (NC) plus fine dans/avec le web sémantique. Nous voulons utiliser le web sémantique pour faire de la négociation (1) et utiliser la négociation quand il s'agit du web sémantique (2).

- (1) Dans la négociation, nous utilisons les technologies du web sémantique, par exemple les métadonnées décrivant une ressource dans différents vocabulaires.
- (2) Lorsque nous voulons disposer de ressources web sémantiques spécifiques, par exemple en demandant une ressource à l'aide du vocabulaire ou un profil.

Le reste de cet article est structuré comme suit : La section 2 présente le problème, des cas d'utilisation le motivant, les questions de recherche et les hypothèses, le cadre de notre recherche et les contributions attendues, suivies de l'état de l'art et des travaux connexes dans la section 3. La section 4 décrit brièvement l'approche proposée, tandis que la section 5 décrit la méthodologie de recherche prévue. Les résultats préliminaires sont décrits dans la section 6, suivis de la conclusion et des travaux futurs dans la section 7.

2 Problème

2.1 Problématique

Une ressource disponible sous un identifiant de ressource uniforme (URI) spécifique peut avoir différentes représentations que nous appelons variantes ou alternatives comme dans [7, section 1.3]. La NC est le mécanisme permettant de choisir la meilleure représentation parmi les variantes disponibles. Le client inclut un ensemble de contraintes dans sa requête tandis que le serveur délivre une représentation sélectionnée lorsqu'elle correspond à ses propres

contraintes [8].

Les questions qui se posent immédiatement sont : quelles sont ces contraintes ? comment les exprimer et comment les faire correspondre ? Au fil des années, les contraintes ont pris de nombreuses formes, en commençant par le type de média, la langue et l'encodage [8], en passant par des contraintes liées à la capacité du dispositif à traiter une certaine représentation [15] jusqu'à être plus complexes pour indiquer une interprétation sémantique [2].

Pour un serveur, outre l'interprétation et la correspondance des contraintes, un autre problème est l'explication du choix, soit de la représentation sélectionnée, soit des alternatives fournies qui ont été jugées suffisamment proches de celle demandée.

2.2 Cas d'utilisation

2.2.1 La négociation du vocabulaire

Chloé gère un portail d'information sur des artefacts anciens (indiquant p.ex. le créateur, la date de création, les matériaux utilisés) dont les données sont récoltées de différentes sources, telles que des musées ou Wikidata. Elle a remarqué que généralement chaque source utilise des vocabulaires différents, parfois personnalisés.

Dernièrement, elle a reçu de nombreuses demandes pour trouver un moyen de rechercher des données dans un vocabulaire spécifique, ou de spécifier les vocabulaires souhaités de manière ordonnée. Par exemple, exposer les données des créateurs en utilisant le vocabulaire FOAF (Friend Of A Friend), Schema.org ou DCMI (Dublin Core Metadata Initiative).

Actuellement, les graphes de données disponibles sur les API utilisent le même type de média : *text/turtle*. L'utilisateur doit interroger manuellement tous les graphes de données pour sélectionner ceux qui utilisent le vocabulaire souhaité.

2.2.2 Négociation des formes RDF

Alexandre est un chercheur qui s'intéresse à l'évolution du chômage des jeunes dans différentes sociétés ; il a besoin de données sous forme de graphes de données RDF. Pour ce faire, il interroge les graphes de données disponibles dans diverses API Web interrogées par le portail de l'université. Ces scénarios sont plausibles :

Scénario 1 - La forme est également importante : Alexandre a besoin d'une représentation qui se conforme à une forme spécifique. Par conséquent, la négociation du vocabulaire n'est pas suffisante car il devrait valider manuellement tous les graphes de données retournés avec les vocabulaires souhaités.

Scénario 2 - Flexibilité vs rigidité : Dans le scénario 1, la négociation peut être rigide dans le cas où Alexandre veut que *toutes* les contraintes soient valides, et préfère ne pas avoir de réponse autrement. Sinon, la négociation peut être flexible dans le cas où il accepte de recevoir une représentation même si elle ne satisfait pas toutes les contraintes.

Scénario 3 - Les contraintes n'ont pas la même importance : Pour Alexandre, toutes les contraintes de forme n'ont pas le même degré d'importance. Il veut donc un

moyen d'exprimer cette importance pour chaque contrainte et d'obtenir la représentation qui minimise le taux de violation en la prenant en compte.

2.3 Questions de recherche et hypothèses

Dans cette recherche, nous cherchons à répondre aux questions suivantes :

- RQ1** Quelles sont les caractéristiques d'un cas d'utilisation de la NC, et comment peut-on les comparer et les classer ?
- RQ2** Comment formaliser de manière uniforme les différents styles et dimensions de la NC ?
- RQ3** Comment utiliser la validation sémantique pour demander la meilleure représentation qui valide partiellement la requête parmi un ensemble de variantes ?
- RQ4** Comment évaluer la faisabilité et la qualité des algorithmes et méthodes proposés ?

Nos hypothèses sont directement dérivées des questions de recherche :

- H1** La création d'une ressource ciblant la documentation sur la NC, en plus de fournir des cas d'utilisation de la NC avec leurs solutions existantes ou potentielles, encouragerait l'utilisation de la NC.
- H2** Les technologies du web sémantique peuvent contribuer au mécanisme de la NC.
- H3** Shapes Constraint Language (SHACL) n'est pas seulement utile pour valider les graphes RDF mais peut être utilisé pour introduire une certaine flexibilité dans le processus de choix de la meilleure représentation.

2.4 Cadre de la recherche

À ce stade de la thèse, nous avons fixé le cadre de la recherche pour inclure la NC basée sur les préférences de l'utilisateur et l'interprétation sémantique, ce qui signifie que, par exemple, les conditions du réseau, les caractéristiques du dispositif, les capacités du dispositif, l'état de la batterie, le coût monétaire sont exclus pour le moment. Cependant, bien que nous nous intéressions au mécanisme de la NC en général, une fois que nous aurons instancié notre modèle général, nous le ferons principalement à l'aide des sources RDF [5].

Ce cadre se manifeste dans la formalisation, nous avons commencé par la classe de document, puis ses sous-classes, cependant nous ne sommes pas intéressés à ce stade par la vitesse des flux de données par exemple. Il convient de mentionner que ce cadre ne nous empêchera pas de spécifier des cas d'utilisation employant le mécanisme de négociation dans la ressource Content Negotiation Theoretical Framework (CNTF) que nous développons dans le contexte de ce travail, par exemple, le cas d'utilisation de la négociation de la fréquence de mise à jour des objets (fraîcheur des données) dans un environnement Web des objets (WoT). Cependant, nous ne les prendrons pas en compte dans la partie formalisation du problème.

2.5 Contributions attendues

Les contributions attendues de cette recherche sont les suivantes :

- C1** Un site web qui répertorie, catégorise et relie les techniques de la NC, les cas d'utilisation et les travaux connexes. En plus d'avoir le rôle de plateforme de diffusion pour nos futures contributions.
- C2** Une formalisation du problème de la NC du général (négociation basée sur les documents) au spécifique (graphe RDF).
- C3** Une solution pour la NC fine en utilisant des langages de validation sémantique, tels que SHACL.
- C4** l'implémentation et la validation de la solution proposée.

3 État de l'art

La NC a été proposée comme une couche essentielle de l'architecture Web depuis le début [3], et a été implémentée dans le protocole HTTP en fournissant les moyens de négocier des variantes par le biais d'en-têtes entre autres : *accept* pour exprimer une contrainte sur le type de média de la représentation, et *accept-language* pour sélectionner la langue préférée [8]. Ainsi qu'un ensemble de codes d'état de réponse à utiliser pour indiquer si une demande HTTP spécifique a été satisfaite. Mais dans la plupart des cas, ces codes sont génériques et ne fournissent aucune explication. Avec HTTP, différents styles de la NC ont émergé, notamment la NC proactive, qui rend le serveur responsable du choix de la meilleure alternative, et la NC réactive, dans laquelle le serveur fournit une liste d'alternatives et c'est au client de choisir la meilleure. Il convient également de mentionner la NC transparente [13], qui permet aux mandataires de choisir au nom du serveur en tirant parti de l'en-tête HTTP *vary* [8]. Notre travail récent détaille davantage cette partie de l'état de l'art [29].

Malheureusement, les en-têtes HTTP de base ne sont pas suffisants, mais le protocole peut être étendu avec des nouveaux en-têtes. Des exemples d'en-têtes personnalisés sont *prefer* pour demander que certains comportements soient utilisés par un serveur [24], *accept-presentation* pour négocier la présentation RDF [17], *accept-schema* pour demander comment la ressource doit être structurée [26]. Cette approche résout le problème mais n'est pas évolutive compte tenu du fait qu'un nouvel en-tête est créé pour chaque nouvelle exigence. De plus, l'interopérabilité n'est pas atteinte puisque le nouvel en-tête doit être connu à l'avance.

Les capacités sont une autre dimension qui doit être négociée. Pour l'agent utilisateur, l'en-tête *user-agent* était utilisé, mais lorsque les appareils mobiles sont introduits, une nouvelle approche a été nécessaire, CC/PP (Composite Capabilities/Preference Profiles) et UAPProfile ont été proposées pour résoudre ce problème [15, 21]. Mais aujourd'hui encore, alors que l'utilisation du Web des objets se développe, la négociation doit tenir compte de leurs limites, comme la faible puissance du processeur et de la batterie.

RDF est destiné à décrire les ressources sur le web en utilisant des vocabulaires et des ontologies. Et pour négocier ces

représentations, le groupe de travail sur l'échange de données a proposé le vocabulaire de profil [2], et des moyens de négocier les profils qui pourraient prendre la forme de ressources dans le langage de contrainte, par exemple SHACL ou ShEx [28, 27, 16, 23]. Cependant, cela manque de flexibilité, par exemple dans le cas où un serveur doit choisir entre deux représentations qui ne valident que partiellement un profil.

Pour répondre aux impacts sur les performances et la confidentialité de l'envoi d'en-têtes qui ne sont pas utilisés de manière fiable dans le traitement d'une requête client, une proposition récente est celle des Client Hints [10]. Un en-tête de réponse *Accept-CH* est introduit que les serveurs peuvent utiliser pour annoncer leur utilisation des en-têtes de requête pour la négociation proactive de contenu. Cependant, à notre avis, l'adoption d'en-têtes contribuant à la NC est insuffisante en raison de l'absence d'une ressource d'orientation. Cette ressource devrait documenter les approches et les caractéristiques de la NC, et les présenter de manière digeste dans le contexte de cas d'utilisation. Ces derniers comprendraient leurs solutions respectives ou potentielles et des pointeurs vers d'autres liens pour une investigation plus approfondie.

4 Approche proposée

Dans cette section, nous discutons de l'approche adoptée pour répondre aux différentes questions de recherche, qui comprend un cadre théorique prenant la forme d'un site web, une formalisation et un algorithme pour augmenter la flexibilité de la NC.

4.1 Cadre théorique de la négociation de contenu

Nous pensons que la NC est l'un des piliers du Web et mérite d'être étudiée plus en profondeur. Nous prévoyons de commencer par une documentation et une présentation bien conçues pour les nouveaux venus qui veulent apprendre la NC, ses caractéristiques et ses cas d'utilisation, ou pour les experts qui veulent se tenir au courant des dernières techniques et technologies utilisées [RQ1]. A cette fin, nous proposons de créer un cadre théorique pour catégoriser et évaluer les différents cas d'utilisation de la NC, et nous voulons matérialiser cela dans une ressource prenant la forme d'un site web que nous présentons dans la section suivante. CNTF (Content Negotiation Theoretical Framework) est le site web que nous développons dans le cadre de ce travail pour permettre la catégorisation des caractéristiques de la NC en différents groupes : style, dimension, etc. CNTF vise à collecter des cas d'utilisation de la NC, à mettre en évidence les solutions existantes si elles sont disponibles, ou suggérer des moyens plausibles de les faire progresser. Ce sont les principaux objectifs de CNTF. Plus tard, elle sera utilisée pour la diffusion de nos nouvelles propositions pour faire avancer la NC. CNTF tente de répondre à ces exigences :

Conception navigable : CNTF doit avoir un design navigable qui inclut la possibilité de passer d'un concept à un

autre, par exemple d'un cas d'utilisation à la dimension ou au style de la NC utilisé via des liens.

Extensible : L'un des principaux contrastes entre un document sur l'état de l'art traditionnel et la ressource CNTF est qu'elle doit être extensible en permettant l'ajout de nouveaux concepts de la NC, par exemple une nouvelle dimension, et être à jour avec les terminologie et les définitions différentes.

Catégorisable : CNTF doit fournir les moyens de catégoriser les différents cas d'utilisation et techniques de la NC pour permettre une évaluation comparative. Elle doit également prévoir un regroupement et une modélisation bien pensés des différents concepts de la NC pour en faciliter la compréhension.

Maintenable : CNTF devrait favoriser la maintenabilité en ajustant le modèle utilisé, par exemple les vocabulaires recommandés par la communauté, et en tenant compte des réactions et des commentaires fournis par les utilisateurs de CNTF pour clarifier et rectifier le contenu.

4.2 Une formalisation ascendante de la NC

Une autre façon d'étudier la NC sémantique est d'avoir plus de précision et de rigueur dans sa formalisation, pour cela nous suggérons une formalisation ascendante de la NC [RQ2]. Bien qu'il y ait eu des tentatives de formalisation de la NC, cela a été fait principalement pour s'adapter à un certain contexte, comme l'adaptation de contenu [19]. Dans notre démarche de formalisation de la NC dans des contextes sémantiques, nous prévoyons d'adopter une approche progressive, depuis la négociation de documents Web de base jusqu'à la négociation de documents de graphes spécialisés en utilisant des langages de validation sémantique, par exemple SHACL.

Le but est de définir de manière formelle ce qu'est une requête client avec ses contraintes, ce qu'un serveur doit fournir comme réponse, quelles sont certaines stratégies pour choisir la représentation à retourner. Ensuite, nous ajoutons progressivement de la complexité, par exemple comment exprimer une redirection lorsque le serveur ne peut pas satisfaire la demande mais peut fournir une URI qui aiderait le client. Et plus tard, nous pouvons définir les protocoles de manière formelle.

4.3 Un pas vers la NC sémantique

Pour résoudre les cas d'utilisation déjà présentés et parvenir à une NC flexible de manière pratique [RQ3, RQ4], nous utilisons SHACL. Plus précisément, nous utilisons l'en-tête *accept-profile* récemment introduit pour demander une variante qui valide un ensemble de contraintes sous la forme de documents SHACL [27]. Un client fait une demande avec un document SHACL. Le serveur, quant à lui, dans la procédure traditionnelle, dispose de l'ensemble des profils correspondant aux variantes. Si une variante est conforme au profil demandé, elle est servie, sinon un code d'état 406 (Not Acceptable), 404 (Not found) ou 300 (Multiple Choices) est renvoyé en fonction de la configuration

du serveur¹. Dans notre approche, nous proposons de valider à la volée le profil demandé avec la liste des variantes disponibles, et de fournir la variante la plus proche si plusieurs valident partiellement les contraintes. Nous avons développé un algorithme simple pour montrer comment cela peut être réalisé : il prend en entrée une liste d'URI de documents SHACL S qui représentent les contraintes du client, chacune avec une valeur numérique optionnelle q_s indiquant la préférence de ce profil. Le serveur dispose d'une liste de graphes de données (variantes) G parmi lesquels il peut choisir. Le résultat de cet algorithme est un graphe de données dont le nombre de violations est minimum. Pour chacun des documents SHACL $s \in S$, nous validons les graphes de données disponibles et enregistrons le nombre de contraintes testées n_c ainsi que les contraintes valides v_c . Ensuite, nous calculons la mesure de validation avec la formule :

$$v_m = \frac{v_c}{\text{Max}(1, n_c)} \times q_s$$

Une fois que chaque paire (document de contraintes, graphe de données) a une mesure de validation, nous livrons celle qui a le meilleur score.

Une autre approche pour augmenter la flexibilité de la NC est l'endiguement des contraintes (constraint containment) [22, 18, 25, 1]. En prenant le document SHACL comme contraintes, nous pouvons dire qu'une forme est contenue dans une autre forme si chaque nœud d'un graphe satisfaisant les contraintes de la première forme satisfait également les contraintes de la seconde.

Considérons trois graphes de formes (SG) :

$$SG_1 = \{S_1, S_2, S_3\}$$

$$SG_2 = \{S_1, S_2, S_3, S_4, S_5\}$$

$$SG_3 = \{S_1, S_3, S_6\}$$

Nous pouvons imaginer trois cas de négociation :

Case 1 : le client demande SG_1 qui est contenu dans SG_2 car tous les S_n de SG_1 sont contenus dans SG_2 .

Case 2 : le client demande SG_2 qui est partiellement contenu dans SG_1 car certains S_n de SG_2 sont contenus dans SG_1 mais SG_1 n'a pas de S_n supplémentaires.

Case 3 : le client demande SG_1 ou SG_3 . Dans les deux cas, nous avons des S_n supplémentaires, SG_1 est partiellement contenu dans SG_3 et SG_3 est partiellement contenu dans SG_1 .

Les comportements suivants du serveur peuvent répondre à ces cas :

Case 1 : Comme la SG demandée est entièrement contenue dans la SG disponible, nous validons les données avec celles demandées et les retournons.

Case 2 : Puisque la SG demandée est partiellement contenue dans la SG disponible, nous pouvons soit valider les formes supplémentaires au graphe de données, soit retourner uniquement le graphe de données avec la SG disponible appliquée avec précision et rappel.

Case 3 : Nous calculons la précision et le rappel entre le graphe de formes demandé et le graphe disponible. Ensuite, nous construisons un nouveau SG incluant uniquement les formes dont le test d'endiguement est positif. Enfin, nous le renvoyons à l'utilisateur avec les graphes de forme originaux et les mesures de précision et de rappel.

5 Méthodologie

La méthodologie adoptée dans l'élaboration de ce travail de doctorat respecte les tâches suivantes :

1. Investigation de l'état de l'art de la recherche relative au problème identifié. Cela inclut l'étude de la littérature sur les techniques de la NC, la formulation et la validation des contraintes dans le domaine du Web sémantique.
2. Formalisation du problème de la NC.
3. Création de la structure et des catégories de CNTF.
4. Collecte des techniques de la NC motivées par des cas d'utilisation et les ajouter à CNTF.
5. Fourniture des algorithmes pour la NC flexible et injection des implémentations dans un espace de test dans CNTF.

Après avoir réalisé un état de l'art approfondi, nous avons choisi de créer un site web qui supporte la catégorisation et l'évaluation comparative des caractéristiques de la NC. Nous avons collecté des cas d'utilisation, et identifié les styles, les dimensions et les exigences de ces cas d'utilisation.

Les techniques de la NC flexible seront d'abord évaluées à l'aide d'un ensemble de données et de profils synthétiques, puis de données réelles pour les travaux futurs. Les performances et l'évolutivité seront mesurées à l'aide du temps de réponse pour un ensemble de graphes et de profils de données prédéfinis. Le temps de réponse sera également mesuré en remplissant les graphes de données avec de plus en plus de triplets, afin de comparer les performances avec un nombre croissant de triplets (par exemple 500, 1000, 1500 triplets). La même technique de mesure sera appliquée pour un nombre croissant de contraintes dans les profils en ajoutant des formes dans le cas de SHACL.

6 Résultats préliminaires

6.1 Le site web CNTF

un site web a été développé², qui, au moment de la rédaction, classe la NC en catégories, un ensemble de cas d'utilisation de la NC, de styles, de dimensions ont été collectés

1. De multiples discussions ont eu lieu au sein du groupe de travail sur l'échange de données pour traiter ces questions, par exemple <https://github.com/w3c/dx-connegp/issues/5>

2. <https://ci.mines-stetienne.fr/cntf>

et ajoutés à la ressource et liés les uns aux autres, la section *updates* dans CNTF fournit la liste des fonctionnalités déjà ajoutées et celles à venir.

6.2 Formalisation de la NC

Une première version de la formalisation de la NC des documents a été réalisée et sera ajoutée à CNTF par la suite. Du côté du serveur, un URI est associé à des documents ayant chacun une valeur de qualité. Pour illustrer, considérons le scénario dans lequel un serveur dispose de deux représentations d'une ressource : d_1 en Turtle avec une préférence de 0,7 et d_2 en texte brut avec une préférence de 0,9. Un client souhaite demander une représentation Turtle. Dans l'ensemble de tous les documents \mathcal{D} , nous avons d_1 et d_2 . La préférence du client pour un document est indiquée par une valeur de qualité q dans l'intervalle $[0, 1]$ ³. Nous définissons l'ensemble \mathcal{C} des préférences comme :

$$c \in \mathcal{C} \mid c : \mathcal{D} \rightarrow [0, 1] \quad (1)$$

En pratique, un client attribue généralement une valeur q à un *type* de documents. Ceci peut être modélisé comme une préférence c telle que $c(d) = q$ pour tous les documents d du type demandé, et $c(d) = 0$ pour tous les autres. Par exemple, si une requête possède l'en-tête `accept: text/turtle; q=0.9`, cela peut être traduit par la préférence $c(d) = 0.9$ pour tous les documents Turtle, et $c(d) = 0$ pour tous les autres.

Nous modélisons le Web comme une fonction de l'ensemble des IRI \mathcal{U} à l'ensemble des contraintes \mathcal{C} , formellement :

$$\mathcal{W} : \mathcal{U} \rightarrow \mathcal{C}$$

Le serveur de notre scénario peut servir deux documents :

$$\mathcal{W}(u) : \begin{cases} d_1 \mapsto 0.7 \\ d_2 \mapsto 0.9 \\ \mathcal{D} \setminus \{d_1, d_2\} \rightarrow 0 \end{cases} \quad (2)$$

Nous modélisons la réponse RES_{ponse} à une requête REQ_{uest} comme un document, en réalité la réponse contient des informations supplémentaires telles que des en-têtes. Dans notre modèle, RES est une fonction définie comme :

$$RES : REQ \rightarrow \mathcal{D} \quad (3)$$

La réponse dépend de la stratégie du serveur. Considérons une requête d'un client vers un IRI u et avec une contrainte c_1 définie comme $\langle u, c_1 \rangle$ dans REQ . Et un serveur qui peut servir une réponse à partir de l'IRI demandé u mais qui a aussi une contrainte c_2 , on peut donc écrire $c_2 = \mathcal{W}(u)$. Nous définissons trois réponses possibles :

$$RES_s : \langle u, c_1 \rangle \rightarrow \underset{|d \in \mathcal{D} \wedge \mathcal{W}(u)(d) \neq 0\}}{argmax}(\{\mathcal{W}(u)(d)\})$$

En utilisant la fonction RES_s , le serveur sert la réponse qui maximise ses contraintes, sans tenir compte des contraintes du client.

$$RES_c : \langle u, c_1 \rangle \rightarrow \underset{|d \in \mathcal{D} \wedge c_1(d) \neq 0 \wedge \mathcal{W}(u)(d) \neq 0\}}{argmax}(\{c_1(d)\})$$

En utilisant la fonction RES_c , le serveur sert la réponse qui maximise les contraintes du client, c'est-à-dire, $RES_{noAnswer} = \emptyset$.

Le serveur n'envoie pas de réponse s'il existe un conflit entre ses contraintes et celles du client.

6.3 Un pas vers la négociation sémantique de contenu

Un autre résultat qui mérite d'être mentionné est une démonstration fonctionnelle de la NC à l'aide de profils qui prennent la forme de documents SHACL. L'implémentation a été faite en utilisant Java, et Spring Framework pour gérer les requêtes et intercepter les en-têtes de requête. Jena Framework a été utilisé pour gérer les graphes RDF, les documents SHACL et la validation. Une fois que nous aurons validé l'évolutivité des résultats, nous ajouterons l'implémentation à CNTF.

7 Conclusions et travaux futurs

La NC est très importante et constitue un mécanisme fondamental du Web. Les avantages et la nécessité d'exploiter la NC sont soulignés par le groupe de travail Spatial Data on the Web [30], ainsi que par le groupe de travail Web Data Best Practices [6] dans leurs documents sur les meilleures pratiques.

Dans cet article, nous analysons l'état des travaux existants en matière de la NC et indiquons une nouvelle direction : l'utilisation de langages de validation pour exprimer des contraintes plus fines. Nous pensons que dans les applications du monde réel, ces contraintes plus fines, lorsqu'elles sont explorées de manière appropriée, rendraient le processus de négociation beaucoup plus flexible. Nous avons proposé la ressource CNTF, la formalisation générale du processus de négociation, et un algorithme pour la négociation de contraintes utilisant SHACL. Notre travail futur se concentrera sur l'enrichissement de CNTF avec plus de cas d'utilisation et la finalisation des caractéristiques pour satisfaire les exigences mentionnées ci-dessus, ainsi que l'extension de la formalisation pour inclure la négociation de documents spécifiques tels que les graphes RDF en utilisant la négociation de profil. Les résultats de cette étude peuvent être utilisés pour valider et affirmer que SHACL peut être utilisé pour introduire de la flexibilité dans le processus de choix de la meilleure représentation.

Nous prévoyons de suivre ces étapes pour nos travaux futurs :

1. Continuer à évaluer et à étendre la négociation de profils.

3. Dans la RFC7231 [8, Section 5.3.1], 0 signifie non acceptable, 0,001 est le moins préféré et 1 est le plus préféré.

2. Enrichir la formalisation pour inclure les solutions proposées.
3. Étudier plus en profondeur l'endiguement des contraintes [22, 18, 25, 1], puis proposer un algorithme qui en tire parti.
4. Prendre un cas d'utilisation réel d'un portail comme Europeana⁴ qui sert de contenu provenant de différentes sources.
5. Explorer le cas d'utilisation de la NC du web des objets/hypermédia dans un environnement de bâtiment intelligent.
6. Examiner l'adaptation de contenu [20].

Références

- [1] A. Abbas, P. Genevès, C. Roisin, and N. Layaïda. SPARQL Query Containment with ShEx Constraints. In *Proc. 21st European Conference on Advances in Databases and Information Systems*, volume 10509 of *LNCS*, pages 343–356. Springer, 2017.
- [2] R. Atkinson and N. Car. The Profiles Vocabulary, W3C Working Group Note 18 December 2019. W3c working group note, W3C, December 18 2019.
- [3] T. Berners-Lee, R. Cailliau, J. Groff, and B. Pollermann. World-Wide Web : The Information Universe. *Electronic Networking : Research, Applications and Policy*, 2(1) :74–82, 1992.
- [4] N. Choudhury. World Wide Web and Its Journey from Web 1.0 to Web 4.0. *Int. Journal of Comp. Sci. and Information Tech.*, 5(6) :8096–8100, 2014.
- [5] R. Cyganiak, D. Wood, and M. Lanthaler. RDF 1.1 Concepts and Abstract Syntax, W3C Recommendation 25 February 2014. W3c recommendation, W3C, February 25 2014.
- [6] B. Farias Lóscio, C. Burle, and N. Calegari. Data on the Web Best Practices, W3C Recommendation 31 January 2017. W3c recommendation, W3C, January 31 2017.
- [7] R. Fielding, J. Gettys, J. Mogul, H. Nielsen, L. Masinter, P. Leach, and T. Berners-Lee. Hypertext Transfer Protocol - HTTP/1.1. RFC 2616, IETF, 1999.
- [8] R. Fielding and J. Reschke. Hypertext Transfer Protocol (HTTP/1.1) : Semantics and Content. RFC 7231, IETF, June 2014.
- [9] R. Fisher, W. Ury, and B. Patton. *Getting to yes : Negotiating agreement without giving in*. Penguin, 2011.
- [10] I. Grigorik and Y. Weiss. HTTP Client Hints. RFC 8942, IETF, February 2021.
- [11] W3C OWL Working Group. OWL 2 Web Ontology Language Document Overview (Second Edition), W3C Recommendation 11 December 2012. W3c recommendation, W3C, December 2012.
- [12] P. Hayes and P. Patel-Schneider. RDF 1.1 Semantics, W3C Recommendation 25 February 2014. W3c recommendation, W3C, February 25 2014.
- [13] K. Holtman and A. Mutz. Transparent Content Negotiation in HTTP. Technical report, IETF, March 1998.
- [14] I. Jacobs and N. Walsh. Architecture of the World Wide Web, Volume One, W3C Recommendation 15 December 2004. W3c recommendation, W3C, December 15 2004.
- [15] G. Klyne, F. Reynolds, C. Woodrow, H. Ohto, J. Hjelm, M. Butler, and L. Tran. Composite Capability/Preference Profiles (CC/PP) : Structure and Vocabularies 1.0, W3C Recommendation 15 February 2004. W3c recommendation, W3C, January 15 2004.
- [16] H. Knublauch and D. Kontokostas. Shapes Constraint Language (SHACL), W3C Recommendation 20 July 2017. W3c recommendation, W3C, July 20 2017.
- [17] M. Lefrançois. RDF presentation and correct content conveyance for legacy services and the web of things. In *Proceedings of the 8th International Conference on the Internet of Things, IOT 2018, Santa Barbara, CA, USA, October 15-18, 2018*, pages 43 :1–43 :8. ACM Press, October 2018.
- [18] M. Leinberger, P. Seifer, T. Rienstra, R. Lämmel, and S. Staab. Deciding SHACL Shape Containment Through Description Logics Reasoning. In *The Semantic Web - ISWC 2020 - 19th ISWC, Athens, Greece, November 2-6, 2020, Proceedings, Part I*, volume 12506 of *LNCS*, pages 366–383. Springer, 2020.
- [19] S. Lerouge. *Personalizing quality aspects for video communication in constrained heterogeneous environments*. PhD thesis, Ghent University, 2006.
- [20] W. Lum and F. Lau. User-Centric Content Negotiation for Effective Adaptation Service in Mobile Computing. *IEEE Trans. on Soft. Eng.*, 29(12) :1100–1111, 2003.
- [21] Open Mobile Alliance. User Agent Profile. Technical report, Open Mobile Alliance, May 2003.
- [22] P. Pareti, G. Konstantinidis, F. Mogavero, and T. Norman. SHACL Satisfiability and Containment. In *The Semantic Web - ISWC 2020 - 19th ISWC, Athens, Greece, November 2-6, 2020, Proceedings, Part I*, volume 12506 of *LNCS*, pages 474–493. Springer, 2020.
- [23] E. Prud'hommeaux, I. Boneva, J. Labra Gayo, and G. Kellogg. Shape Expressions Language 2.1, Final Community Group Report 8 October 2019. W3c community group report, W3C, 2019.
- [24] J. Snell. Prefer Header for HTTP. RFC 7240, IETF, 2014.
- [25] S. Staworko and P. Wiecek. Containment of Shape Expression Schemas for RDF. In *Proc. 38th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pages 303–319. ACM Press, 2019.

4. <https://www.europeana.eu/>

- [26] L. Svensson. An http Header for Metadata Schema Negotiation. In *W3C Workshop on Smart Descriptions & Smarter Vocabularies (SDSVoc)*. W3C, November 2016.
- [27] L. Svensson, R. Atkinson, and N. Car. Content Negotiation by Profile, W3C Working Draft 26 November 2019. W3C Working Draft, W3C, November 26 2019.
- [28] L. Svensson, R. Verborgh, and H. Van de Sompel. Indicating, Discovering, Negotiating, and Writing Profiled Representations. Internet draft, IETF, March 2021.
- [29] Y. Taghzouti, A. Zimmermann, and M. Lefrançois. Négociation de contenu sur le web : un état de l’art. In *Journées Francophones d’Ingénierie des Connaissances (IC 2022) @ Plate-Forme Intelligence Artificielle (PFIA 2022)*, 2022. to appear.
- [30] L. van den Brink, P. Barnaghi, J. Tandy, G. Atemezing, R. Atkinson, B. Cochrane, Y. Fathy, R. García-Castro, A. Haller, A. Harth, K. Janowicz, S. Kolozali, B. van Leeuwen, M. Lefrançois, J. Lieberman, A. Perego, D. Le Phuoc, B. Roberts, K. Taylor, and R. Troncy. Best practices for publishing, retrieving, and using spatial data on the web. *Semantic Web Journal*, 10(1) :95–114, 2019.