

Négociation de contenu sur le Web: un état de l'art

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

L'ouverture et l'accessibilité du Web a largement contribué à son adoption à l'échelle mondiale. L'identifiant uniforme de ressource (URI) est la pierre angulaire de l'identification des ressources sur le Web. Une ressource sur le Web peut être décrite de nombreuses façons, ce qui peut rendre difficile pour un utilisateur de trouver une représentation adéquate. Cela a motivé une recherche fructueuse sur la négociation de contenu (NC) afin de satisfaire les besoins de l'utilisateur de manière efficace. Notre objectif de recherche est de présenter un état de l'art qui inclut (1) une identification des caractéristiques des scénarios de la NC (styles, dimensions et moyens de transmission des contraintes), (2) une comparaison et une classification de quelques contributions existantes, (3) des cas d'utilisation pas encore couverts dans l'état de l'art, (4) une suggestion de directions de recherche pour les travaux futurs. Les résultats de l'état de l'art montrent que le problème de la NC est pertinent et loin d'être résolu.

Mots-clés

Négociation de contenu, Web sémantique, contrainte.

Abstract

The openness and accessibility of the Web has contributed greatly to its worldwide adoption. Uniform Resource Identifier (URI) is the cornerstone of resource identification on the Web. A resource on the Web can be described in many ways. The large number of ways to describe a resource can make it difficult for a user to find an adequate representation. This situation has motivated fruitful research on content negotiation (CN) to satisfy user requirements efficiently and effectively. We focus on the topic of CN, and our goal is to present a comprehensive state of the art that includes (1) identifying the characteristics of CN scenarios (styles, dimensions, and means of conveying constraints), (2) comparing and classifying some existing contributions, (3) identifying use cases that the current state of CN struggles to address, (4) suggesting research directions for future work. The results of the state of the art show that the problem of CN is relevant and far from being solved.

Keywords

Content negotiation, semantic web, constraint.

1 Introduction

Ouvert, distribué, accessible et hétérogène sont quelques-unes des caractéristiques fondamentales du Web [5]. Le fait que chacun puisse accéder au Web de n'importe où dans le monde a grandement contribué à son développement et à son enrichissement grâce à son ouverture. Cependant, cela a eu l'effet indésirable d'avoir une abondance de ressources Web, et par conséquent des difficultés à fournir le meilleur contenu pour chaque utilisateur; un exemple simple est celui de deux personnes parlant des langues différentes accédant à la même ressource. Le serveur avec la ressource devrait être capable de fournir à chaque utilisateur 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 abordée plus tard dans le document Architecture of the World Wide Web comme l'un des composants essentiels de la conception Web [13, 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 [10, p. 1]. Appliquée au Web, elle devient alors la "négociation de contenu" (NC) : 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 [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, puisque 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].

1.1 Énoncé du problème

Les ressources sur le Web sont accessibles via des identifiants uniformes de ressources (URI). Une ressource peut avoir différentes alternatives que nous appelons variantes comme dans [7]. Pour demander une variante spécifique, un utilisateur utilise la NC. Ainsi, la NC est le mécanisme permettant de choisir la meilleure variante parmi un ensemble d'alternatives d'une ressource disponible sur le Web sous

un URI. La demande envoyée par un client contient un ensemble de contraintes qui permettent au serveur de fournir une réponse adéquate. Le traitement de la demande par le serveur peut varier en fonction du style et de la dimension de la NC utilisée, ainsi que de la technique de transmission des contraintes et du protocole employé.

1.2 Portée de l'étude

Nous avons utilisé la stratégie suivante pour identifier les caractéristiques de la NC ainsi que les contributions disponibles faisant l'objet de notre classification dans notre état de l'art. Nous avons utilisé le mot clé "content negotiation" pour rechercher des articles dans Google Scholar¹, The DBLP Computer Science Bibliography² et Semantic Scholar³. Nous avons choisi des articles, des rapports et des études empiriques pertinents parmi les ressources collectées en nous basant sur la compatibilité du problème étudié avec notre énoncé de problème présenté dans la section 1.1. En outre, nous avons suivi les citations et les références afin de considérer les ressources supplémentaires qui citent ou sont citées par les ressources pertinentes ci-dessus. Sept contributions ont été retenues pour être classées dans cet article. Une version à jour du classement est disponible dans la ressource *CNTF* sur laquelle nous travaillons⁴.

1.3 Problèmes et études connexes

Des efforts ont été faits pour rassembler les techniques et les caractéristiques disponibles de la NC. Toutefois, à notre connaissance, cela n'a pris la forme que d'une section de travaux connexes dans les articles traitant de la NC, comme [23, Section 5] ou de la documentation de pages Web, comme dans les Mozilla Developer Network Web Docs⁵. Ces efforts ne contiennent que les dimensions générales de la NC. Par exemple ils ne mentionnent pas la dimension de capacité que nous aborderons plus tard, ou ne traitent que certains styles de la NC.

Un autre problème connexe est la recherche d'informations personnalisée, qui vise à aider les utilisateurs à trouver des informations parmi la myriade de ressources disponibles sur le Web. Cependant, contrairement au domaine traditionnel de la recherche d'information, elle prend en compte les préférences de l'utilisateur et l'historique de ses interactions avec le système dans le but principal d'augmenter la satisfaction de l'utilisateur. Une enquête a été menée et a proposé une classification de ces systèmes [11].

1.4 Contribution de l'article

La NC a donné lieu à des résultats de recherche fructueux ces dernières années. Cependant, il n'existe pas d'état de l'art approfondi sur ce sujet de recherche. À notre connaissance, nous fournissons le premier état de l'art des approches et des caractéristiques de la NC existantes. Nous identifions les caractéristiques de chaque contribution et les

catégorisons afin de classer les contributions existantes sur la NC. Sur la base de cet état de l'art, nous proposons des cas d'utilisation qui ne sont pas encore bien traités et suggérons des directions pour les travaux futurs.

1.5 Structure de l'article

Le reste de cet article est organisé comme suit. La section 2 présente la terminologie de la NC. La section 3 passe en revue les caractéristiques de la NC et les regroupe en catégories, ce qui nous permet dans la section 4 de fournir une analyse comparative d'une liste de contributions dans ce domaine. Enfin, nous concluons avec la présentation de deux cas d'utilisation que la NC, dans son état actuel, n'est pas en mesure de traiter pleinement. Nous suggérons une orientation future dans la section 5.

2 Terminologie

Cette section présente la terminologie de base de la NC.

Agent Utilisateur (User-Agent) : est un navigateur ou une application mobile par lequel l'utilisateur communique avec le serveur [7].

Client : est le code d'application qui veut accéder aux ressources du serveur et établit des connexions afin d'envoyer des requêtes [7].

Serveur : est un programme d'application qui accepte les connexions afin de répondre aux demandes en renvoyant des réponses. Un programme donné peut être capable d'être à la fois un client et un serveur. Notre utilisation de ces termes fait uniquement référence au rôle joué par le programme pour une connexion particulière, plutôt qu'aux capacités du programme en général. De même, tout serveur peut agir en tant que serveur d'origine, proxy, passerelle ou tunnel, changeant de comportement en fonction de la nature de chaque requête [7].

Ressource : est un objet ou service de données réseau qui peut être identifié par une URI [7].

Variante : est l'une des représentations associées à une ressource à un instant donné [7].

Capacité : est un attribut d'un expéditeur ou d'un destinataire (souvent le destinataire) qui indique une capacité à générer ou à traiter un type particulier de contenu de message [16].

Ressource négociable : est une ressource de données à laquelle est associé plusieurs représentations (variantes). La sélection d'une variante appropriée pour la transmission dans un message est accomplie par la négociation du contenu entre l'expéditeur et le destinataire [16].

Style de négociation : est la manière dont la négociation est menée et de la partie de la NC qui choisit la variante à sélectionner.

Dimension de négociation : également appelée *contraintes* ou *préférences* : indiquent les contraintes prises en compte lors de la sélection de la meilleure représentation.

1. <https://scholar.google.com/>

2. <https://dblp.org/>

3. <https://www.semanticscholar.org/>

4. <https://w3id.org/cntf/classification>

5. https://developer.mozilla.org/en-US/docs/Web/HTTP/Content_negotiation

3 Caractéristiques de la NC

Nous divisons les caractéristiques de la NC en style, dimension et moyen de transmission des contraintes.

3.1 Dimension de négociation du contenu

Les dimensions de la NC sont les principaux facteurs de différenciation entre un ensemble de variantes. Nous disons que deux représentations alternatives varient en fonction d'une ou plusieurs dimensions de la NC⁶.

Type de média : Si un utilisateur a une préférence ou que le client ne peut traiter qu'un type de média spécifique⁷, il peut exprimer cela avec HTTP à l'aide de l'en-tête *Accept*.

Langue : Si un utilisateur a une préférence pour une langue particulière, il peut l'exprimer avec HTTP à l'aide de l'en-tête *Accept-Language*.

Encodage : Si un client souhaite que la réponse soit codée à l'aide d'un algorithme de compression particulier. Avec HTTP, il peut l'exprimer à l'aide de l'en-tête *Accept-Encoding*. Les valeurs acceptées sont les suivantes : *gzip, compress, deflate, br, identity* ou *.

Char-Set : Avant que *UTF-8* ne soit largement pris en charge, un client pouvait négocier l'encodage des caractères qu'il supportait. Avec HTTP, on l'exprime à l'aide de l'en-tête *Accept-Charset*.

Capacité : En général, un attribut qui définit les capacités et les préférences du matériel ou du logiciel d'un récepteur⁸. Un exemple est la possibilité de traiter les couleurs, les niveaux de gris ou simplement le noir et blanc.

Versión : Une ressource pouvant avoir différentes représentations de l'état par lequel elle est passée dans le temps, [6] a introduit un cadre pour négocier les états de la ressource par le biais de la négociation par date, en ajoutant l'en-tête *Accept-Datetime*.

Systèmes de référence de coordonnées : Les objets sur la terre peuvent être localisés à l'aide d'un système de référence de coordonnées. Différents utilisateurs peuvent être intéressés par une représentation de ces objets dans différents systèmes : un système local, régional ou mondial, par exemple WGS84, ETRS89 ou Lambert93.

Autorisation : Les données sur le Web, comme nos comportements dans le monde réel, pourraient être soumises à une réglementation. Les ressources pourraient être limitées, par exemple, pour des raisons de confidentialité. Les utilisateurs peuvent vouloir demander des données uniquement si elles sont conformes à certaines règles et s'ils disposent des autorisations appropriées.

Vocabulaire : Resource Description Framework (RDF) est un langage général permettant de représenter les informations sur le Web. Ces informations sur le Web peuvent être décrites à l'aide de différents vocabulaires. Les utilisateurs

peuvent vouloir demander des représentations décrites à l'aide d'un certain vocabulaire, par exemple des données sur une personne décrites par FOAF ou vCard.

Profil OWL : Un profil OWL est une version réduite de *OWL full* qui échange une certaine puissance d'expression contre une efficacité de raisonnement. Pour cette raison, les utilisateurs peuvent souhaiter demander une ontologie dans un profil OWL spécifique tel que *OWL 2 EL*.

Régime d'inférence : Diverses normes du W3C, dont RDF et OWL, fournissent des interprétations sémantiques pour les graphes RDF qui permettent de déduire des déclarations RDF supplémentaires à partir d'assertions explicites. Un utilisateur peut vouloir demander des données conformes à une forme e.g. SHACL/ShEx mais après avoir utilisé un régime d'inférence spécifique tel que RDF Schema, pour valider également les triplets inférés.

Résumé : L'abondance et la longueur des représentations sur le Web rendent leur utilisation de plus en plus difficile pour les humains. Un utilisateur peut vouloir demander un résumé d'une représentation présentant des caractéristiques spécifiques au lieu de recevoir la représentation originale complète.

Mise en page : La mise en page en général fait référence à la manière dont le texte, les images, etc. sont organisés. Un utilisateur peut vouloir demander une représentation qui respecte une mise en page donnée, comme la version minifiée d'un fichier JS, ou le style de citation d'une bibliographie.

Exactitude : La mesure est un processus qui utilise des nombres pour décrire une quantité physique. Les unités de mesure fournissent des normes afin que les nombres dans nos mesures se réfèrent à la même chose. Un utilisateur peut vouloir demander une représentation avec une précision ou une unité de mesure particulière.

Profil : Une définition d'un profil est "une description des contraintes structurelles et/ou sémantiques sur les représentations des ressources qui s'appliquent en plus des contraintes intrinsèquement indiquées par leur type de média" [24]. Un utilisateur peut souhaiter recevoir une représentation uniquement si elle est conforme à un profil particulier, tel que le profil d'application DCAT [23].

3.2 Style de la NC

Dans notre étude, nous avons identifié six styles de la NC : *proactif, réactif, transparent, actif, conditionnel, adaptatif*. Chacun d'entre eux présente des compromis en termes d'applicabilité et d'aspect pratique⁹. Les sous-sections suivantes présentent une brève description de chaque style, ainsi que certains avantages et inconvénients génériques associés à son utilisation.

3.2.1 Proactive

On parle de négociation proactive (aussi appelée négociation pilotée par le serveur) lorsque les préférences de la NC

6. Notez que cette liste n'est pas exhaustive et que les dimensions ne sont pas orthogonales

7. Liste des types de média : <https://www.iana.org/assignments/media-types/media-types.xhtml>

8. Un exemple de fichier Nokia CC/PP <http://nds1.nds.nokia.com/uaprof/N6230ir200.xml>

9. Il est important de mentionner que les styles de NC ne sont pas mutuellement exclusifs.

sont envoyées par le client dans une requête afin d'encourager un algorithme situé sur le serveur à sélectionner la représentation préférée. La sélection est basée sur les représentations disponibles pour une réponse (les dimensions selon lesquelles elle peut varier, comme la langue, le codage du contenu, etc.) par rapport aux diverses informations fournies dans la requête [8].

Avantages

- Le serveur évite les allers-retours car le client envoie les préférences au serveur qui fait sa meilleure estimation et l'envoie avec la réponse.
- Le serveur n'a pas besoin de décrire l'algorithme de sélection au client pour faire un choix.

Désavantages

- Il est impossible pour le serveur de déterminer avec précision ce qui serait "le mieux" pour un utilisateur donné, car cela nécessiterait une connaissance complète des capacités de l'agent utilisateur et de l'utilisation prévue de la réponse.
- Le fait que l'agent utilisateur doit décrire ses capacités dans chaque requête est un risque potentiel pour la vie privée de l'utilisateur.
- Il complique la mise en œuvre d'un serveur d'origine et des algorithmes de sélection de variantes.
- Il limite la réutilisation des réponses pour la mise en cache partagée.

3.2.2 Réactive

Si le serveur reçoit une demande ambiguë, il envoie une liste des différentes variantes dont il dispose. L'agent utilisateur peut faire le choix s'il a une connaissance suffisante des préférences de l'utilisateur final. Sinon, il affiche la liste des liens pour que l'utilisateur final fasse son choix [8].

Avantages

- Le serveur est incapable de déterminer la capacité de l'agent utilisateur via les en-têtes, ce qui implique plus de confidentialité.
- Le cache peut être utilisé pour réduire la surcharge du réseau.

Désavantages

- Latence due à l'aller-retour pour sélectionner la représentation.

3.2.3 Transparente

La NC transparente est appelée ainsi parce qu'elle rend visibles aux parties intermédiaires (entre le serveur d'origine et l'agent utilisateur) toutes les variantes qui existent au sein du serveur d'origine et leur donne la possibilité de choisir la meilleure représentation en leur nom. La NC transparente est une combinaison de la NC proactive et réactive. Dans la NC réactive, lorsqu'un cache est fourni sous la forme d'une liste de représentations disponibles de la réponse et que la partie intermédiaire (un proxy cache) a bien compris les dimensions de la variance, alors la partie intermédiaire devient capable d'effectuer une NC proactive au nom du serveur d'origine pour les demandes ultérieures sur cette ressource [12, 21, 8].

Avantages

- La réduction du temps de réponse et de la consommation de bande passante en raison de la distribution du travail de négociation qui serait autrement requis du serveur d'origine.
- Le gain du délai de la deuxième demande de négociation réactive lorsque la partie intermédiaire utilise le cache pour pouvoir deviner correctement la bonne réponse.

Désavantages

- Ce style suppose une mise en cache maximale des ressources, ce qui, en pratique, n'est vrai que pour le contenu statique et non crypté.

3.2.4 Conditionnelle

La réponse qu'un serveur donne à une demande dans un style de la NC conditionnelle consiste en un corps composé de plusieurs parties séparées par des limites. Les parties sont rendues de manière sélective en fonction des paramètres de l'agent utilisateur. Cela peut prendre la forme de parties contenant différentes variantes de la ressource, par exemple avec des types de médias distincts¹⁰, ou de parties contenant des portions d'une représentation, par exemple certaines pages d'un document PDF [9, 8]¹¹.

Avantages

- Réduire le nombre de requêtes émises à une seule qui obtient une réponse composée de plusieurs parties, chaque partie contenant une variante.
- Sélectionner uniquement une partie d'une représentation.

Désavantages

- Ne passe pas à l'échelle si le nombre de variantes ou la taille des variantes sont importants.

3.2.5 Active

Le serveur dans la NC active répond avec une réponse qui contient un script. Le script effectue des demandes supplémentaires (plus spécifiques) en fonction des caractéristiques de l'agent utilisateur [8].

Avantages

- La réduction de l'interaction de l'utilisateur en automatisant l'émission de demandes supplémentaires.
- La fourniture d'une représentation personnalisée qui correspond aux capacités et des besoins de l'agent utilisateur.

Désavantages

- Le besoin d'envoyer plusieurs requêtes pour construire la représentation finale.
- L'introduction de menaces potentielles dues à l'exécution de scripts, par exemple, un attaquant de type man-in-the-middle peut intercepter ou réécrire la réponse pour y inclure du code malveillant.

10. <https://docs.marklogic.com/9.0/guide/rest-dev/bulk>

11. Dans le cas de parties contenant différentes variantes, ce style est différent du style réactif : dans le premier cas, nous envoyons les variantes réelles séparées par des frontières, alors que dans le second, seuls les URI des variantes disponibles sont envoyés, et des requêtes supplémentaires sont nécessaires pour les récupérer

- L'interdiction du contenu actif par défaut dans les versions les plus récentes des navigateurs en raison des vulnérabilités mentionnées ci-dessus.

3.2.6 Adaptative

Le serveur dans la NC adaptative répond avec une représentation qui a subi un processus d'adaptation. L'adaptation peut être effectuée en interne par le serveur ou en utilisant un autre service pour effectuer cette tâche, par exemple en effectuant une transformation qui nécessite beaucoup de puissance de traitement [19]. L'adaptation est réussie si la représentation finale livrée est plus compréhensible pour l'utilisateur en fonction de son contexte.

Avantages

- Augmenter le taux de satisfaction des contraintes.

Désavantages

- Le processus de pré-adaptation du contenu ne passe pas à l'échelle.

3.3 Transmission de contraintes dans la NC

Dans un processus de la NC, le client doit transmettre au serveur la dimension de négociation ainsi que sa valeur à prendre en compte pour le processus de sélection de la meilleure variante à fournir. Deux techniques principales ont émergé et sont largement utilisées pour effectuer la transmission des contraintes : L'approche *HTTP headers* et l'approche *URL*. Les sous-sections suivantes décrivent chacune de ces deux techniques.

3.3.1 En-têtes HTTP

Les en-têtes HTTP sont des éléments essentiels du protocole HTTP qui permettent la transmission d'informations supplémentaires par le client (en-têtes de requête) et le serveur (en-têtes de réponse). Cette section montrera comment les en-têtes HTTP sont utilisées pour transmettre des contraintes. Il est important de préciser que le protocole CoAP (Constrained Application Protocol) implémente également la NC, mais uniquement pour les type de média via l'option *accept* [22, Section 5.10.4].

La ressource CNTF contient plus de détails ¹².

Accept : en-tête de requête qui indique le type de média que l'utilisateur préfère. Le serveur sélectionne une variante et informe le client de son choix avec les en-têtes de réponse *Content-Type* [8, Section 5.3.2].

Accept-Language : une en-tête de requête qui indique la langue exprimée par le langage naturel et la locale que l'utilisateur préfère. Le serveur sélectionne une variante et informe le client de son choix avec les en-têtes de réponse *Content-Language* [8, Section 5.3.5].

Accept-encoding : une en-tête de requête qui indique l'encodage. Typiquement un algorithme de compression que le client préfère. Le serveur sélectionne une proposition et informe le client de son choix avec les en-têtes de réponse *Content-Encoding* [8, Section 5.3.4].

Accept-Crs : une en-tête de requête qui indique le CRS que l'utilisateur préfère. Le serveur sélectionne une proposition et informe le client de son choix avec les en-têtes de réponse *Content-Crs* ¹³.

Accept-Presentation : une en-tête de requête qui indique la présentation que l'utilisateur préfère. Le serveur sélectionne une proposition et informe le client de son choix avec les en-têtes de réponse *Content-Presentation* [18].

3.3.2 Basée sur l'URL

Les URI ne fournissent pas seulement un moyen simple et extensible d'identifier des ressources sur le Web, mais ils pourraient aussi être utilisés pour transmettre des contraintes afin de guider la sélection d'une variante préférée. Dans cette section, nous présentons trois façons de les utiliser.

Archival Resource Key (ARK) : L'ARK est un schéma d'identification pour un identifiant persistant des objets d'information [17]. En utilisant la partie facultative "Qualifier", il est possible de créer une sorte de point d'entrée de service qui permet à un ARK de prendre en charge l'accès aux variantes (versions, langues, formats) des composants en utilisant le caractère "." (point) après la partie Nom d'un ARK. Par exemple, l'URI `https://api.istex.fr/ark:/67375/6GQ-MLC8GRWC-5` liste toutes les variantes et la variante PDF se trouve à l'URI `https://api.istex.fr/ark:/67375/6GQ-MLC8GRWC-5/fulltext.pdf`.

Extension de l'URL (suffixe de correspondance de motifs) : cette approche est similaire à l'approche ARK, qui consiste à utiliser l'URI avec une extension principalement pour demander le type de média des points de terminaison de l'API. L'URI `http://myapi.example.com/account/123.json` fournirait une représentation du compte 123 dans un format json.

Query String Arguments (QSA) : Une chaîne de requête est une partie d'une URL qui attribue des valeurs à des paramètres spécifiques. Les QSA sont généralement utilisées pour fournir des informations supplémentaires à un serveur. L'une d'entre elles concerne les préférences pour sélectionner une variante appropriée ¹⁴.

4 Classification de certaines contributions existantes

Le tableau 1 présente notre effort pour rassembler quelques contributions connues dans la littérature ayant en commun leur utilisation de la NC. Chaque contribution est présentée avec sa référence, sa date de publication ainsi que ses caractéristiques respectives selon notre catégorisation présentée précédemment.

Composite Capabilities / Preferences Profile (CC/PP) et WAP UAProf sont des descriptions des capacités de péri-

13. <https://github.com/opengeospatial/conneg-by-crs/>

14. e.g. http://linked.data.gov.au/dataset/gnaf/address/GAACT714845933?_view=ISO19160&_format=text/turtle

12. <https://w3id.org/cntf>

Ref	Date	Style	Dimension	Transmission	Protocole
[4]	2001	Proactive	Capacité	en-tête	HTTP
[20]	2003	Adaptative	Capacité	QSA	HTTP
[25]	2018	Réactive	Type de média	ARK	HTTP
[14]	2018	Réactive	Version	en-tête	HTTP
[18]	2018	Proactive, Adaptative	Présentation	en-tête	HTTP, CoAP
[1]	2018	Conditionnelle	Media type	en-tête	HTTP
[2]	2022		Type de média	en-tête	CoAP

TABLE 1 – Contributions utilisant la NC (triées par date de publication) et leurs caractéristiques (Blanc : non connu).

phériques et des préférences des utilisateurs. [4] décrit une implémentation de la NC HTTP qui les utilise pour fournir la meilleure variante. Le style proactif de la NC est utilisé et la dimension est *capacité*. Les en-têtes sont utilisés pour transmettre les contraintes et le protocole est HTTP.

L'article [20] présente un moteur de décision capable de déterminer les décisions d'adaptation optimales à partir de l'interpolation d'informations contextuelles situationnelles, par exemple la capacité du dispositif. HTTP a été utilisé en intégrant le *userid* dans l'URL pour identifier l'utilisateur final.

Istex est une archive scientifique française [25]. Les utilisateurs ont la possibilité d'avoir la représentation dans plusieurs formats et pour cela, comme mentionné sur le site, un ARK est utilisé. Si on demande la ressource sans spécifier le type de média, on reçoit un fichier json décrivant les variantes existantes et on peut donc considérer qu'il s'agit d'une négociation réactive. La dimension est le type de média et la transmission dans ARK en utilisant le protocole HTTP.

Les archives sur le Web jouent un rôle majeur en fournissant une image reflétant l'état du Web à un moment donné. La contribution [6] implique la demande du client dans le processus d'agrégation Memento au delà de la spécification de l'URI de la ressource originale et d'une date comme décrit dans [14] en utilisant l'en-tête *prefer*. Par conséquent, la dimension de la NC utilisée est la version, les en-têtes pour le transport des contraintes et HTTP comme protocole. De plus, comme l'agrégateur renvoie un ensemble d'archives et que le client peut potentiellement manipuler la réponse pour émettre une autre requête, nous jugeons que le style de NC est réactif.

Un scénario décrit dans la contribution [18, Section 3.2] est qu'un client demande à un serveur la représentation d'une ressource et souhaite que la réponse soit encodée selon une présentation RDF spécifique, ce qui est fait en incluant des métadonnées dans l'en-tête de la requête. Et parce que cette contribution est principalement destinée aux dispositifs contraints, le style proactif + adaptatif est préféré. L'auteur précise que, même si cette méthode a été implémentée en HTTP, elle pourrait facilement être définie comme des options CoAP équivalentes.

En utilisant le serveur MarkLogic [1], un développeur d'applications REST utilise le style de la NC conditionnelle. Un client recevrait une réponse avec un corps contenant plusieurs parties séparées par un délimiteur à sélectionner.

Cette méthode est principalement utilisée pour sélectionner la dimension du type de média et utilise HTTP et les en-têtes pour transmettre cette contrainte. En outre, la spécification de l'Open Connectivity Foundation (OCF) [2] comporte aussi les moyens de négociation du type de média à l'aide du protocole CoAP.

5 Conclusions et travaux futurs

La NC est un mécanisme fondamental du Web. Dans cet état de l'art, nous avons présenté une analyse des caractéristiques des scénarios de la NC existants, notamment les styles, les dimensions, les moyens de transmission des contraintes et les protocoles de la NC. Ces caractéristiques peuvent être utilisées pour classer les contributions existantes comme indiqué dans le tableau 1.

Cependant, certains cas d'utilisation n'ont pas encore de solution satisfaisante, par exemple le cas de la négociation de vocabulaire où un utilisateur veut avoir un moyen de demander des représentations utilisant un vocabulaire spécifique. Par exemple, demander que les données du créateur utilisent le vocabulaire FOAF (Friend Of A Friend), Schema.org ou DCMI (Dublin Core Metadata Initiative). Si les graphes de données disponibles sur le serveur utilisent le même type de média, par exemple : *text/turtle*, le client doit interroger manuellement tous les graphes de données pour sélectionner ceux qui utilisent le vocabulaire souhaité. Une autre limitation est mise en évidence avec le cas d'utilisation de la négociation des formes RDF, où un utilisateur a besoin d'une représentation conforme à une forme spécifique, donc même la négociation du vocabulaire n'est pas suffisante car le client devrait valider manuellement tous les graphes de données retournés. Dans ce cas, la négociation peut être rigide dans le cas où l'utilisateur veut que *toutes* les contraintes soient valides, et préfère ne pas avoir de réponse autrement. En revanche, la négociation peut être flexible dans le cas où l'utilisateur accepte de recevoir une représentation même si elle ne satisfait pas toutes les contraintes. Et enfin, ces contraintes de forme peuvent ne pas avoir le même degré d'importance. L'utilisateur peut donc vouloir un moyen d'exprimer cette importance pour chaque contrainte et d'obtenir la représentation qui en tient compte.

Une direction plausible est l'utilisation des langages de validation pour exprimer des contraintes plus fines, par exemple le langage SHACL du Web sémantique. Plus précisément, l'utilisation de l'en-tête récemment intro-

duit *accept-profile* pour demander une variante qui valide un ensemble de contraintes sous la forme de documents SHACL [23].

Références

- [1] REST Application Developer's Guide. Technical report, MarkLogic Corporation, May 2019.
- [2] OCF Core specification 2.2.5. Technical report, Open Connectivity Foundation, January 2022.
- [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] Mark H. Butler. Implementing content negotiation using CC/PP and WAP UAProf. *HP Laboratories Technical Report HPL*, (190), 2001.
- [5] 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.
- [6] Herbert Van de Sompel, Michael L. Nelson, and Robert Sanderson. HTTP framework for time-based access to resource states - memento. RFC 7089, 2013.
- [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] Roy Fielding, Yves Lafon, and Julian Reschke. Hypertext transfer protocol (HTTP/1.1) : range requests. RFC 7233, IETF, 2014.
- [10] R. Fisher, W. Ury, and B. Patton. *Getting to yes : Negotiating agreement without giving in*. Penguin, 2011.
- [11] M Rami Ghorab, Dong Zhou, Alexander O'connor, and Vincent Wade. Personalised information retrieval : survey and classification. *User Modeling and User-Adapted Interaction*, 23(4) :381–443, 2013.
- [12] K. Holtman and A. Mutz. Transparent Content Negotiation in HTTP. RFC 2295, IETF, March 1998.
- [13] I. Jacobs and N. Walsh. Architecture of the World Wide Web, Volume One. W3c recommendation, W3C, December 15 2004.
- [14] Mat Kelly, Sawood Alam, Michael L. Nelson, and Michele C. Weigle. Client-Assisted Memento Aggregation Using the Prefer Header. In *Proc. of WADL'18*, Fort Worth, TX, June 2018.
- [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, W3C, January 15 2004.
- [16] Graham Klyne. Protocol-independent content negotiation framework. Technical report, IETF, 1999.
- [17] John A. Kunze and Emmanuelle Bermès. The ARK Identifier Scheme. Internet draft, IETF, 2008.
- [18] 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.
- [19] Tayeb Lemlouma and Nabil Layaïda. Universal profiling for content negotiation and adaptation in heterogeneous environments. In *W3C Workshop on Delivery Context*, pages 4–5, 2002.
- [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] Srinivasan Seshan, Mark Stemm, and Randy H Katz. Benefits of transparent content negotiation in http. In *Proceedings of the IEEE Globcom 98 Internet Mini-Conference*. Citeseer, 1998.
- [22] Zach Shelby, Klaus Hartke, and Carsten Bormann. The Constrained Application Protocol (CoAP). Technical report, Internet Engineering Task Force, June 2014.
- [23] L. Svensson, R. Atkinson, and N. Car. Content Negotiation by Profile. W3C Working Draft, W3C, November 26 2019.
- [24] L. Svensson, R. Verborgh, and H. Van de Sompel. Indicating, Discovering, Negotiating, and Writing Profiled Representations. Internet draft, IETF, March 2021.
- [25] Pascale Viot and Nicolas Thouvenin. Istex : une nouvelle corde à son ark. *Arabesques*, (88) :18–19, 2018.