

Projet Knowledge Management



LAVEDER Mark-Ylann
[Lien vers le dépôt GitHub](#)

18/06/2022

Sommaire

1	Choix du domaine	1
2	Taxonomie	1
3	Ajout d'instances	2
3.1	Réseau	2
3.2	Client	2
3.3	Serveur	2
4	Ajout de prédicats	2
5	Changements additionnels	3
6	Requêtes SPARQL	4

Table des figures

1	Taxonomie simplifiée d'une relation client-serveur	1
2	Sous-graphe de la taxonomie simplifiée	2
3	Graphe de la taxonomie mis à jour	3

1 Choix du domaine

J'ai choisi le domaine de la relation client-serveur car il s'agit d'un de mes champ d'action dans mon travail quotidien. Etant employé comme apprenti ingénieur DevOps, je suis amené à travailler et à résoudre des problématiques sur les deux plans.

Le contexte de mon entreprise n'a pas été retenu et seules quelques technologies sont citées.

2 Taxonomie

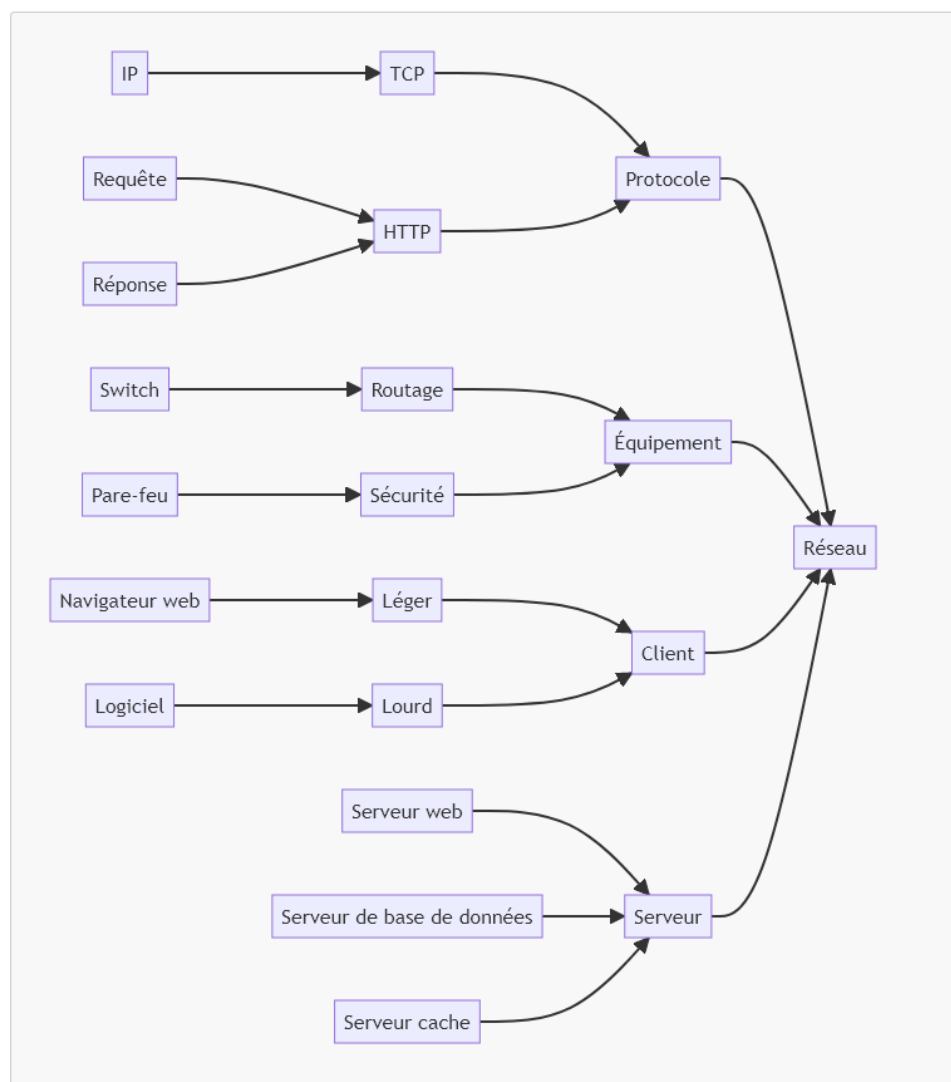


FIGURE 1 – Taxonomie simplifiée d'une relation client-serveur

3 Ajout d'instances

3.1 Réseau

- IP : IPv4, IPv6
- Requête : GET, POST
- Réponse : 200, 404

3.2 Client

- Navigateur : Chrome, Firefox

3.3 Serveur

- Serveur Web : Nginx, IIS
- Serveur de base de données : MySQL, SQLServer
- Serveur cache : Redis, Memcached

4 Ajout de prédicats

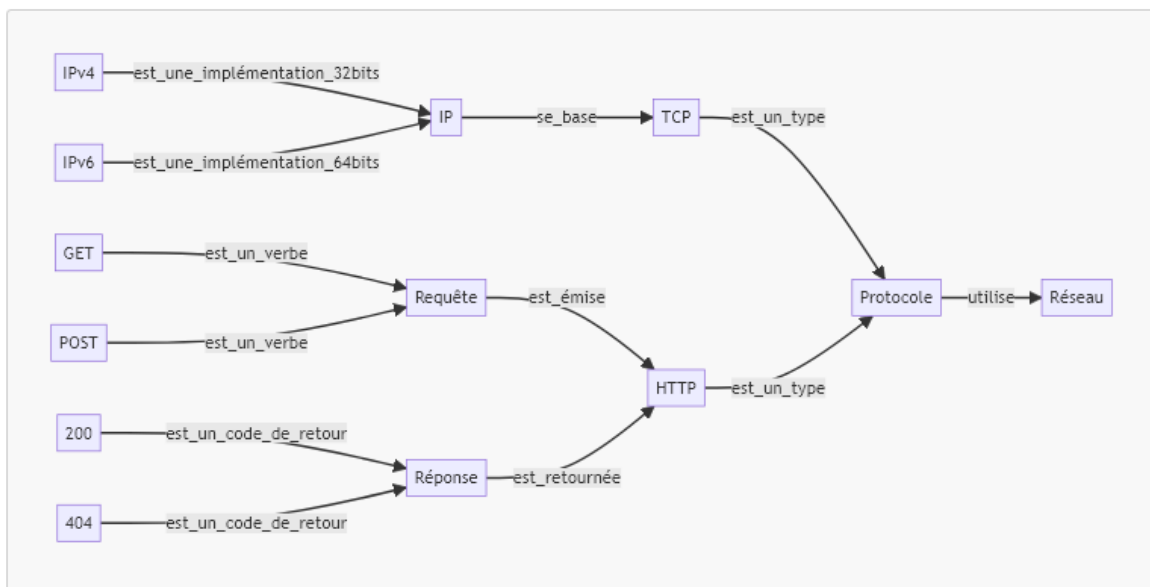


FIGURE 2 – Sous-graphe de la taxonomie simplifiée

5 Changements additionnels

La représentation graphique de la taxonomie comporte des prédicats entre tous les noeuds.

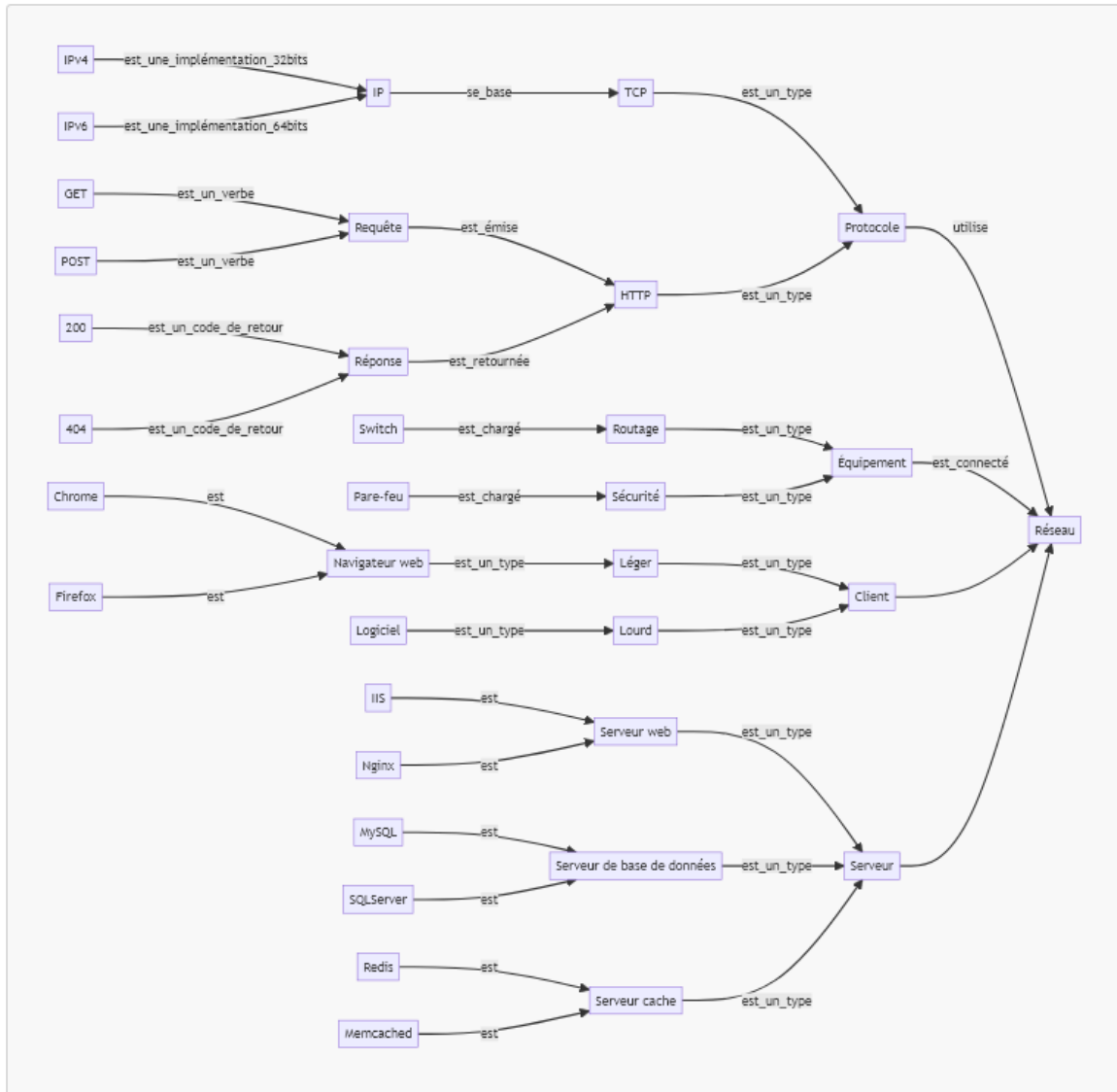


FIGURE 3 – Graphe de la taxonomie mis à jour

Ecriture en Turtle de la taxonomie.

Transformation des instances 200 et 404 en ‘OK’ et ‘Not Found’ avec un lien ‘a un code retour’.

Ajout d’un noeud Fonction, lié à Equipement et à ses sous-classes.

6 Requêtes SPARQL

Requête		Résultat
Sélectionner les instances de ServeurWeb		
<pre>SELECT ?serveurWeb WHERE { ?serveurWeb rdf:type <http://example.com/ServeurWeb>. }</pre>	<pre>----- serveurWeb ----- <http://example.com/Nginx> <http://example.com/IIS> ----- </pre>	
Décrire l'instance "Nginx"		
<pre>DESCRIBE <http://example.com/Nginx></pre>	<pre>(graph (triple <http://example.com/Nginx> rdf:type owl:NamedIndividual) (triple <http://example.com/Nginx> rdf:type <http://example.com/ServeurWeb>) (triple <http://example.com/Nginx> <http://example.com/version> "1.23.4"))</pre>	
Demander s'il existe une instance de Client		
<pre>ASK { ?nav rdf:type <http://example.com/NavigateurWeb>. }</pre>	<pre>yes</pre>	

Construire un graphe des ServeurWeb et leur version	
<pre> 1 CONSTRUCT { 2 ?serveurWeb <http:// example.com/version> ? version. 3 } 4 WHERE { 5 ?serveurWeb <http:// example.com/version> ? version . 6 ?serveurWeb rdf:type <http ://example.com/ ServeurWeb>. 7 }</pre>	<pre> (graph (triple <http://example.com/ Nginx> <http://example.com /version> "1.23.4") (triple <http://example.com/ IIS> <http://example.com/ version> "2.34.5"))</pre>
Insérer une nouvelle instance de ServeurWeb	
<pre> 1 INSERT DATA { 2 <http://example.com/Apache > rdf:type <http:// example.com/ServeurWeb >. 3 <http://example.com/Apache > <http://example.com/ version> "3.45.6". 4 }</pre>	<pre> SPARQL update executed correctly</pre>
Supprimer une instance de ServeurWeb	
<pre> 1 DELETE { 2 <http://example.com/Apache > rdf:type <http:// example.com/ServeurWeb >. 3 } 4 WHERE { 5 <http://example.com/Nginx> rdf:type <http:// example.com/ServeurWeb >. 6 }</pre>	<pre> SPARQL update executed correctly</pre>