



HAL
open science

Temporalité et représentation des connaissances pour un corpus en histoire : application à la correspondance d'Henri Poincaré

Nicolas Lasolle

► To cite this version:

Nicolas Lasolle. Temporalité et représentation des connaissances pour un corpus en histoire : application à la correspondance d'Henri Poincaré. 2022. hal-03681513

HAL Id: hal-03681513

<https://hal.univ-lorraine.fr/hal-03681513>

Preprint submitted on 30 May 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Temporalité et représentation des connaissances pour un corpus en histoire : application à la correspondance d’Henri Poincaré

Nicolas Lasolle^{1,2}

30 mai 2022

¹ Université de Lorraine, CNRS, Université de Strasbourg, AHP-PreST,
F-54000 Nancy, France

² Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
`nicolas.lasolle@univ-lorraine.fr`

Résumé

L’étude et la valorisation des travaux d’Henri Poincaré (1854-1912) a mené à la création d’un corpus construit autour de documents de natures diverses (articles, livres, rapports, lettres, etc.). Un intérêt particulier est notamment porté à l’étude de sa correspondance qui rassemble des échanges d’ordres scientifique, administratif et privé. Des technologies du Web sémantique ont été mises en œuvre pour représenter les connaissances de ce corpus et créer des outils les exploitant : le modèle RDF, le langage de représentation de connaissances RDFS et le langage d’interrogation SPARQL. La représentation actuelle ne tient pas compte de la notion de temporalité qui intervient pourtant à plusieurs reprises dans ce corpus. Différents travaux ont proposé des méthodes variées pour associer des éléments temporels aux faits décrits dans une base RDF. Ce rapport présente les problématiques de représentation de connaissances temporelles rencontrées dans ce corpus. Au travers d’exemples, il compare les méthodes de représentation et ontologies existantes et détaille la mise en œuvre de l’approche retenue afin d’intégrer ces éléments temporels. En particulier, un mécanisme de transformation de requêtes SPARQL est utilisé pour simplifier la recherche d’informations au sein d’un graphe RDF temporel.

Introduction

Jules Henri Poincaré est né à Nancy, France, en 1854 et mort à Paris en 1912. Il est considéré comme l’un des derniers grands savants universels de par les contributions majeures qu’il a apportées dans plusieurs disciplines. C’est principalement pour ses apports en mathématiques (formes automorphes, topologie) et en physique (problème des trois corps) qu’il a acquis une telle

renommée. Il a également travaillé à la vulgarisation scientifique et a contribué à la philosophie des sciences comme l'attestent ses trois ouvrages publiés au début du XX^e siècle : *La Science et l'Hypothèse* (POINCARÉ, 1902), *La Valeur de la Science* (POINCARÉ, 1905) et *Science et Méthode* (POINCARÉ, 1908). Durant sa carrière, Henri Poincaré a été très impliqué au sein de nombreuses institutions et sociétés savantes. Il a tenu un rôle dans des sociétés françaises telles que *l'Académie des sciences* ou le *Bureau des longitudes* mais également dans des sociétés internationales comme la *Société hollandaise des sciences de Harleem*, *l'Académie des sciences hongroise* ou la *Société américaine de philosophie*. Au sein de ces différentes institutions, il tenait des statuts variés allant du membre correspondant (comme ce fut le cas pour plusieurs institutions étrangères) au rôle de président (le *Bureau des longitudes* (en 1899, 1909 et 1910) et la *Société astronomique de France* (1901-1903)).

La correspondance d'Henri Poincaré est constituée d'environ 2200 lettres et rassemble des échanges d'ordres scientifique, administratif et privé. Ce corpus se présente comme une source d'informations importante pour les historiens qui s'intéressent à la période. L'étude et la valorisation de ce corpus est un projet de longue date qui a conduit à la publication de plusieurs volumes thématiques. Par exemple, un volume est dédié à la correspondance de jeunesse d'Henri Poincaré (ROLLET, 2017) lorsqu'il était en formation à l'École polytechnique puis à l'École des mines (1873-1878). Un autre volume regroupe les échanges avec le mathématicien suédois Gösta Mittag-Leffler¹, échanges réguliers s'étalant durant une grande partie de la carrière académique d'Henri Poincaré (NABONNAND, 1998). Plusieurs sites Web se sont succédés pour présenter la correspondance et les travaux de ce scientifique.

Accessible depuis septembre 2018, le dernier site en date² a été créé et est maintenu grâce au gestionnaire de contenus *Omeka S* (BOULAIRE et CARABELLI, 2017). Ce système permet à des institutions de publier et rendre accessibles des corpus de documents numérisés. Il est particulièrement adapté pour la valorisation de collections issues de musées, bibliothèques ou lieux d'archives. Le site est hébergé par Huma-Num (LARROUSSE et MARCHAND, 2019) qui propose des services adaptés aux projets d'humanités numériques. Sur cette plate-forme, une lettre est associée à une numérisation du document original³, à une transcription⁴, à un apparat critique ainsi qu'à un ensemble de méta-données descriptives. Celles-ci permettent à la fois la description physique de la lettre (expéditeur, destinataire, date de rédaction, etc.) et de son contenu (thèmes abordés, personnes et institutions citées, etc.). En complément de l'installation d'Omeka S, des technologies du Web sémantique ont été mises en œuvre afin de proposer des outils d'édition de données et de recherche d'informations aux

1. Gösta Mittag-Leffler (1846-1927) est un mathématicien suédois qui est notamment reconnu pour avoir introduit un théorème (portant désormais son nom) lié à la représentation des fonctions méromorphes par des séries de fractions rationnelles. En 1882, il crée la revue *Acta Mathematica* où publieront plusieurs mathématiciens de l'époque comme Georg Cantor, Sophie Kowalevski et Henri Poincaré.

2. <http://henripoincare.fr>.

3. Certains ne sont pas disponibles à cause des droits d'auteurs.

4. Environ 40% des lettres sont associées à une transcription en XML ou L^AT_EX.

utilisateurs. Les méta-données associées aux lettres sont éditées en utilisant le modèle RDF, modèle s'appuyant sur un graphe orienté et étiqueté. Le langage SPARQL permet la formulation de requêtes complexes exploitant les liens entre les ressources et les connaissances de l'ontologie. Par exemple, des historiens pourraient vouloir retrouver les lettres envoyées à des membres de l'Académie des sciences entre 1880 et 1885 et traitant d'optique. Il n'est pas possible de formuler ce type de requêtes avec les modules d'Omeka S qui proposent néanmoins un outil de recherche plein texte appelé Solr⁵, accessible sur le site de la correspondance.

Les données de ce corpus ont été structurées en utilisant une ontologie définie au sein du laboratoire AHP-PRéST⁶. La représentation ne tient pas compte de la validité temporelle des faits décrits dans la base RDF. Pourtant, cette information est importante, particulièrement dans le contexte de l'étude d'un corpus historique. Par exemple, il peut être intéressant de représenter le fait qu'Henri Poincaré ait été membre de l'Académie des sciences. Le triplet RDF correspondant, `(henriPoincaré membre academieDesSciences)`, devrait être considéré comme valide à partir du 31 janvier 1887, date de son élection au sein de l'institution⁷. Intégrer la notion de temporalité dans les graphes RDF n'est pas une tâche aisée et ce domaine de recherche a suscité un intérêt important de la part de la communauté. Différentes problématiques ont été mises en avant dans ces travaux. Cet article considère le problème de représentation de connaissances correspondant à l'association d'un moment ou intervalle temporel à un fait représenté en RDF. L'objectif de ce travail est donc de proposer un modèle permettant de représenter un fait avec une temporalité et d'interroger ces faits avec le langage SPARQL en considérant différents types de requêtes selon que l'on s'intéresse ou non à la temporalité. En particulier, la représentation proposée devra être facilement intégrée à la base RDF actuelle qui contient un nombre important de triplets (environ 230 000). Un autre objectif est de pouvoir mettre en place des raisonnements entre les éléments temporels associés aux faits de la base de connaissances.

La suite de l'article présente un rappel des technologies du Web sémantique utilisées (section 1). Les problématiques de temporalité dans le traitement de corpus historiques sont ensuite détaillées (section 2). En particulier, cette section présente plusieurs limitations dans la représentation des connaissances de ce corpus, au travers de différents exemples de description de faits nécessitant l'ajout d'une composante temporelle. Une synthèse des approches existantes pour représenter les données temporelles au sein d'une base RDF est fournie (section 3). La section 4 présente la mise en œuvre de certaines de ces approches pour la base de données du corpus d'Henri Poincaré. Un mécanisme de transformation de requêtes SPARQL a été adapté pour simplifier l'interrogation de graphes RDF temporels (section 5).

5. <https://lucene.apache.org/solr/>.

6. Archives Henri-Poincaré, Philosophie et Recherche sur les Sciences et les Technologies.

7. Henri Poincaré est élu dans la section de géométrie.

```

    <letter11 sentBy henriPoincaré>
    <letter11 sentTo göstaMittagLeffler>
    <letter11 quotes charlesHermite>
    <letter11 hasTopic ‘‘Géométrie non-euclidienne’’>
    <letter11 hasWritingDate 1882-07-27>

```

FIGURE 1 – Un exemple de triplets décrivant en partie une lettre.

1 Préliminaires sur le Web sémantique

Cette section présente les technologies du Web sémantique utilisées pour représenter les données de ce corpus : le modèle RDF, les langages de représentation de connaissances RDFS et OWL et le langage d’interrogation de graphes SPARQL.

1.1 Le modèle RDF

Resource Description Framework (RDF (LASSILA, SWICK et al., 1998)) est un modèle de représentation de connaissances fondé sur l’utilisation de graphes orientés et étiquetés. C’est un standard développé et maintenu par le *World Wide Web Consortium* (W3C) et qui est fortement utilisé par la communauté. Un graphe RDF est composé de trois types de nœuds : des *ressources nommées*, des *ressources anonymes* et des *littéraux*. Une *ressource nommée* est identifiée par une *Internationalized Resource Identifier* (IRI) et permet de décrire une classe (e.g. **Person**, **Mathematician**, **Letter**, etc.), une propriété (e.g. **sentBy**, **sentTo**, etc.) ou une instance (e.g. **henriPoincaré**, **letter11**, etc.)⁸. Une *ressource anonyme* représente une ressource qui n’est pas explicitement identifiée (*noeud vide*). Une telle ressource est désignée par l’utilisation d’un point d’interrogation préfixant un nom de variable (e.g. **?x**, **?person**, etc.). Un littéral correspond à une valeur constante d’un type donné (**entier**, **chaîne de caractères**, **date**, etc.).

Il est possible de définir des relations entre les nœuds du graphe par l’utilisation de propriétés décrivant les ressources les composant. Ces relations sont caractérisées par des triplets de la forme *< sujet prédicat objet >*. Le sujet représente la ressource (nommée ou anonyme) à décrire. Le prédicat est une propriété qui décrit cette ressource. L’objet est la valeur associée à la propriété et peut être une ressource nommée, une ressource anonyme ou un littéral. La figure 1 donne un exemple de triplets RDF associés à une lettre de la correspondance d’Henri Poincaré.

Il existe différentes syntaxes permettant de créer des bases de données au format RDF. Durant les années qui ont suivi la création de RDF, la syntaxe XML était la plus fréquemment utilisée mais les choses ont évolué depuis. D’autres syntaxes, comme la syntaxe Turtle (CAROTHERS et PRUD’HOMMEAUX, 2014), sont désormais couramment utilisées pour stocker des bases de données RDF.

8. Dans un souci de lisibilité, les ressources nommées ne sont pas représentées en utilisant des IRI complètes dans ce document.

Dans ce document, une syntaxe abstraite, proche de la syntaxe Turtle sera utilisée pour représenter les triplets RDF.

1.2 RDF Schema

RDF schema (RDFS (BRICKLEY et GUHA, 2014)) est un langage de représentation de connaissances qui étend le modèle RDF. Plusieurs propriétés sont utilisées pour structurer les ressources : `rdfs:subclassof` (resp. `rdfs:subpropertyof`) permet de créer une hiérarchie entre des classes (resp. propriétés). Par exemple, le triplet $\langle \text{Letter } \text{rdfs:subclassof } \text{Document} \rangle$ indique que le concept de `Letter` est plus spécifique que le concept de `Document`. `rdfs:domain` (resp. `rdfs:range`) s'applique à une propriété et permet d'ajouter une contrainte à propos du type de la ressource se trouvant à la position de *sujet* (resp. *objet*) au sein d'un triplet. Dans la suite de cet article, des abréviations seront utilisées pour ces différentes propriétés : `a` pour `rdf:type`, `subc` pour `rdfs:subclassof`, `subp` pour `rdfs:subpropertyof`, `domain` pour `rdfs:domain` et `range` pour `rdfs:range`.

Un graphe RDF constitue une base de connaissances à partir de laquelle de nouvelles connaissances peuvent être engendrées. La déduction RDFS désigne l'application de règles d'inférence RDFS afin de dégager de nouvelles connaissances. Ce mécanisme est utilisé pour mener des raisonnements au sein des graphes RDF. Soit un graphe composé des triplets $\langle l \text{ sentTo } \text{henriPoincaré} \rangle$, $\langle \text{sentTo } \text{range } \text{Letter} \rangle$ et $\langle \text{Letter } \text{subc } \text{Document} \rangle$. Considérez les règles r_1 et r_2 :

$$\frac{\langle x \text{ } p \text{ } y \rangle \langle p \text{ range } R \rangle}{\langle y \text{ a } R \rangle} \quad r_1 \qquad \frac{\langle x \text{ a } C \rangle \langle C \text{ subc } D \rangle}{\langle x \text{ a } D \rangle} \quad r_2$$

En appliquant r_1 , il est possible de déduire que la ressource l est de type `Letter` et, sachant cela, il est désormais possible de déduire qu'elle est également de type `Document` par l'application de r_2 .

Des propriétés supplémentaires sont également introduites par RDFS pour décrire les ressources RDF : `rdfs:label` permet d'associer à une ressource un nom qui soit compréhensible par les humains. De la même façon, la propriété `rdfs:comment` permet la définition de la description d'une ressource.

Il existe d'autres langages de représentation de connaissances pour RDF tels que OWL.

1.3 OWL

Web Ontology Language (OWL (ANTONIOU et VAN HARMELEN, 2004)) est un langage de représentation de connaissances qui se décline en 3 versions, chacune étant une extension de la précédente :

OWL Lite : inclut la définition de cardinalités minimales, de hiérarchies et la mise en place de contraintes simples. Cette version est fondée sur la logique de description $\mathcal{SHIF}(\mathcal{D})$;

OWL DL : propose un niveau d'expressivité conséquent tout en garantissant la complétude et la décidabilité des calculs. Cette version est la plus couramment utilisée et est fondée sur la logique de description $\mathcal{SROIQ}(\mathcal{D})$;

OWL Full : fournit une plus grande liberté mais est indécidable et n'est pas supportée par les moteurs de raisonnement OWL actuels.

De nombreuses propriétés sont introduites par OWL pour décrire les classes et propriétés et ainsi structurer des ontologies. Parmi celles-ci, nous pouvons citer des propriétés liées à la notion d'égalité (`owl:equivalentClass`, `owl:equivalentProperty`, `owl:differentFrom`, etc.), à la symétrie (`owl:inverseOf`) et à la conjonction de classes (`owl:intersectionOf`). OWL DL et OWL Full fournissent des éléments supplémentaires permettant de raisonner sur les classes. Ainsi, il est possible de définir l'union (`owl:unionOf`), le complémentaire (`owl:complementOf`) et la disjonction (`owl:disjointWith`) de classes.

1.4 SPARQL

SPARQL est le langage recommandé par le W3C pour interroger des graphes RDF (PRUD'HOMMEAUX, 2008). Une forme courante de requêtes SPARQL est constituée d'une clause SELECT contenant une ou plusieurs ressources anonymes suivie par une clause WHERE composée d'un *patron de graphe* divisé en un ou plusieurs *patrons de triplet* et d'une possible clause FILTER qui permet d'ajouter des contraintes pour les valeurs littérales associées aux propriétés. Considérons la requête informelle suivante :

$$Q = \left\{ \begin{array}{l} \text{« Donner les lettres rédigées par Henri Poincaré} \\ \text{entre 1885 and 1890 et ayant} \\ \text{pour thème la géométrie non-euclidienne »} \end{array} \right.$$

Cette requête peut s'exprimer de la façon suivante en utilisant le langage SPARQL :

$$Q = \left\{ \begin{array}{l} \text{SELECT ?l} \\ \text{WHERE \{?l a letter . ?l sentBy henriPoincaré .} \\ \text{?l subject "Géométrie non-euclidienne" .} \\ \text{?l hasWritingDate ?y .} \\ \text{FILTER(YEAR(?y) >= 1885 AND YEAR(?y) <= 1890)\}} \end{array} \right.$$

2 Ontologie et représentation des connaissances pour un corpus en histoire

Cette section vise à présenter les problématiques liées à la temporalité dans la représentation de connaissances historiques.

2.1 Temporalité et histoire

En histoire, la temporalité joue un rôle clé dans la compréhension et la représentation des événements. En effet, il convient de considérer la dimension diachronique dans le travail des historiens. Ces derniers sont amenés à introduire

des faits et les lier au sein de discours cohérents (PROST, 1996). Dans ce contexte, l'historien a besoin d'établir des chronologies, qu'il doit nuancer en fonction des objets d'études et des questions pour lesquelles il essaye d'apporter des éléments de réponse. Il détermine des périodes construites à partir de dates identifiées comme des *points de rupture*. Ces marqueurs sont arbitraires et il convient de considérer ce point pour appréhender les travaux des historiens (CARLYLE, 1830). C'est un travail d'objectivation qui varie sensiblement selon les problématiques et les objets d'étude. Par exemple, imaginez un chercheur s'intéressant à l'histoire sociale et politique de la France du XIX^e siècle et un second chercheur étudiant l'histoire scientifique de ce siècle. Si certains événements sont considérés dans les deux cas, les dates-clés retenues pourraient être différentes. L'historien de la politique pourrait proposer une périodisation s'appuyant sur les changements de régimes : le Premier Empire (1804-1814) la monarchie constitutionnelle (1815-1848), la Deuxième République (1848-1852), le Second Empire (1852-1870) et la Troisième République (1870-1940). L'historien des sciences aurait plutôt intérêt à mettre en avant des tournants scientifiques : la création de l'homéostasie par Claude Bernard (1850), la mise au point du vaccin contre la rage par Louis Pasteur (1885), la découverte du polonium et radium par Marie et Pierre Curie (1898), etc. (TATON, 1961). Les temporalités sont donc plurielles et les périodes proposées correspondent à des constructions historiques visant à fournir un ensemble de repères communs pour appréhender les liens qui peuvent exister entre présent, passé et futur (KOSELLECK, 2002).

C'est un enjeu majeur que de tenir compte de ces éléments temporels lors de la représentation de connaissances issues de corpus historiques. Ces problématiques de représentation de connaissances temporelles sont abordées dans la suite de cette section. Bien que les exemples s'appuient sur le corpus d'Henri Poincaré, ces problématiques ne lui sont pas spécifiques.

2.2 Une ontologie pour l'histoire

Au sein du laboratoire AHP-PreST, plusieurs travaux sont dédiés à l'étude et à la valorisation de l'œuvre d'Henri Poincaré. Au travers des recherches autour de ses publications scientifiques, il est possible de retracer ses découvertes et ses apports pour les différentes disciplines qu'il affectionnait. Ce travail permet notamment de valoriser les travaux de certains de ses confrères qui menaient des recherches sur des thèmes liés. De plus, Henri Poincaré a rédigé de nombreux rapports et comptes-rendus dans le cadre de ses fonctions au sein de diverses institutions et sociétés savantes.

Un intérêt tout particulier est également porté à l'étude de sa correspondance. Ce corpus regroupe environ 2200 lettres échangées avec plus de 400 interlocuteurs français ou étrangers. Pourquoi étudier cette correspondance ? Plusieurs aspects sont à considérer afin d'apporter des éléments de réponse à cette question. Tout d'abord, l'étude d'une correspondance telle que celle d'Henri Poincaré permet d'apporter des éléments de contexte liés à l'élaboration de théories scientifiques. En effet, à la fin du XIX^e siècle, la lettre était un vecteur non négligeable de l'information scientifique (GILROY et VERHOEVEN, 2000). Au

cours de sa carrière, Henri Poincaré a tenu une correspondance active avec nombre de ses confrères français et étrangers. Par exemple, sa correspondance avec le mathématicien suédois Gösta Mittag-Leffler s'étale durant l'intégralité de sa carrière et permet de retracer la genèse de certains de leurs travaux respectifs. L'étude de la correspondance d'Henri Poincaré peut également éclairer le fonctionnement de différentes institutions et sociétés savantes de son époque de par les échanges réguliers qu'il tenait avec plusieurs autres membres. Explorer cette correspondance permet aussi de mieux comprendre le contexte politique, social et culturel de la France de la fin du XIX^e siècle. Enfin, ce corpus présente Henri Poincaré sous un jour différent de ce qui est mis en avant par l'étude de ses travaux et ainsi introduire de nouveaux éléments biographiques. Ces échanges révèlent plusieurs points de la personnalité de Poincaré reliés à sa méthode de recherche, ses relations, sa vie privée, etc.

Les données du corpus d'Henri Poincaré ont été structurées en s'appuyant sur une ontologie⁹ définie au sein du laboratoire AHP-PreST. Celle-ci a pour objectif de représenter des connaissances liées à des corpus historiques, en modélisant des documents (articles, livres, rapports, lettres, etc.), des personnes et institutions, des lieux, etc.

Cette ontologie présente plusieurs limitations et fait actuellement l'objet d'une restructuration. Le modèle souffre d'ambiguïtés et de redondances et, de plus, il est devenu nécessaire de proposer un alignement plus pertinent avec les ontologies externes : dcterms¹⁰, bibo¹¹, bio¹², rel¹³, etc. L'amélioration puis la publication de cette ontologie permettrait à la communauté scientifique de la réutiliser pour d'autres corpus en histoire. L'une des principales problématiques rencontrées lors de ce travail de restructuration concerne la prise en compte de la validité temporelle des faits, non abordée dans les précédentes versions de l'ontologie. Ces problématiques de représentations temporelles sont détaillées dans la section suivante.

2.3 Faits et temporalité

Plusieurs faits issus du corpus de la correspondance seraient décrits de façon plus pertinente en incluant des notions temporelles. Une première problématique concerne la description des adresses de résidence associées aux personnes. Dans la version actuelle de l'ontologie, la propriété `livesAt` est parfois utilisée pour associer un lieu à une personne. Cependant, la sémantique de cette propriété n'est pas clairement définie. Cela correspond-il à un lieu de vie « principal », que l'on pourrait interpréter comme le lieu où la personne a résidé durant la majeure partie de sa vie ? Ou bien est-ce le lieu de résidence au moment du décès de la personne ? Dans certains cas, cela peut également correspondre au

9. Le terme ontologie fait référence à l'ensemble des concepts et relations utilisés pour représenter les connaissances d'un domaine. Dans le cas des travaux présentés dans ce document, cette ontologie est formalisée avec le langage RDFS.

10. <https://www.dublincore.org/specifications/dublin-core/dcmi-terms/>

11. <http://www.bibliontology.com/>

12. <https://vocab.org/bio/>

13. <https://vocab.org/relationship/>

seul lieu de vie connu qui est associé à la personne. Dans un contexte d'histoire des sciences, il est nécessaire de considérer des connaissances parfois incomplètes à cause du manque de ressources liées à un certain contexte. Dans le cadre de ce corpus, il est fréquent qu'une adresse renseignée corresponde à une résidence principale hypothétique en s'appuyant sur le lieu de travail associé à la personne. Il apparaît clairement un manque ici pour interpréter l'adresse associée à la propriété *livesAt* et il est nécessaire de mener une réflexion pour proposer un ensemble de propriétés mieux structuré. Il serait également nécessaire de pouvoir associer plusieurs lieux à une personne et de pouvoir les distinguer sans ambiguïté. Ces problématiques imposent d'ajouter une composante temporelle à ces relations entre personnes et lieux.

Une autre problématique concerne la représentation des personnes et relations entre elles. Ce terme fait référence à des liens divers qui peuvent être ou non limités dans le temps. En effet, certains liens familiaux sont vrais de façon permanente (ou selon un autre point de vue, en considérant l'intervalle temporel durant lequel les deux personnes sont vivantes). Par exemple, le fait qu'Henri Poincaré a pour mère Eugénie Launois reste valide quelle que soit la période temporelle concernée par le discours. D'autres formes de relations sont quant à elles limitées dans le temps. C'est par exemple le cas pour la relation de mariage, qui possède une date de début et une date de fin. Le fait qu'Henri Poincaré ait été marié à Jeanne Louise Poulain d'Andecy est valide à partir du 20 avril 1881, date de la cérémonie de mariage, et se termine avec sa mort, le 17 juillet 1912. Une autre forme de connaissances, pour le moment non représentées, concerne les relations de travail qui pourraient être désignées par le terme *collègue*. Au cours de sa carrière, Henri Poincaré a été membre de plusieurs institutions. De plus, il est également intervenu en tant qu'enseignant dans différentes formations proposées par divers établissements (École polytechnique, École des mines, Université de Paris, etc.). Il a donc côtoyé de nombreuses personnes qui pourraient être qualifiées de collègues. Cependant, chacune de ses relations existe pour une période de temps spécifique. Par exemple, Henri Poincaré a côtoyé de nombreuses personnes à la Faculté des sciences de Paris au sein de laquelle il a occupé diverses chaires : « mécanique physique et expérimentale » (1884-1886), « physique mathématique et calcul des probabilités » (1886-1896), « astronomie mathématique et mécanique céleste » (1896-1912). Il a ainsi pu côtoyer des personnes telles que Gaston Darboux, Émile Picard ou Joseph-Alfred Serret. Cependant, ces différentes relations sont limitées dans le temps, selon les interventions des uns et des autres dans les différents établissements. Il convient donc de considérer cet aspect temporel lors de l'utilisation des propriétés sous-jacentes.

Un autre objectif est de pouvoir considérer une temporalité comme une réunion d'intervalles. Par exemple, Henri Poincaré a tenu différentes positions au sein du *Bureau des longitudes*¹⁴. Membre à partir de 1893, il fut ensuite

14. Cette société savante française a été créée en 1795 dans l'objectif affiché de reprendre le contrôle des mers en améliorant le système de détermination des longitudes (SCHIAVON et ROLLET, 2017). Rapidement, les disciplines traitées par le Bureau des Longitudes s'étendent et incluent notamment la géodésie, la mécanique céleste, la cosmologie, la métrologie, etc. Cette institution a fonctionné comme une société savante parallèle à l'Académie des sciences.

nommé secrétaire pour les années 1896 et 1897. Il fut également président du Bureau de janvier 1899 à janvier 1900 et une seconde fois entre février 1909 et décembre 1910 (SCHIAVON et ROLLET, 2017). Ainsi, une personne s'intéressant à l'histoire de cette institution pourrait rechercher le moment où Henri Poincaré en était le président. Une telle recherche doit retourner une temporalité t correspondant à une réunion finie d'intervalles temporels disjoints : $t = [18/01/1899, 10/01/1900] \cup [17/02/1909, 28/12/1910]$.

Au-delà de la seule intégration d'éléments temporels, un autre objectif est de pouvoir mettre en place des raisonnements afin de dégager des connaissances à partir des données éditées du corpus. Dans le cadre du corpus de la correspondance, ceux-ci peuvent notamment être utilisés pour estimer la date de rédaction de certaines lettres ou la date de publication de certains articles ou rapports. Par exemple, si une lettre de la correspondance ne présente pas de date de rédaction explicite, mais que le correspondant félicite Henri Poincaré pour son mariage, il est possible de déduire que la lettre a été rédigée après le 20 avril 1881. Si, dans cette même lettre, le correspondant annonce à Henri Poincaré sa venue à Paris pour l'exposition internationale d'Électricité¹⁵, l'exposition ayant eu lieu entre le 10 août 1881 et le 20 novembre 1881, il devient possible d'associer la lettre à une date de rédaction comprise dans l'intervalle [20 avril 1881, 20 novembre 1881]. Par la suite, peut être que d'autres éléments pourraient conduire à un intervalle plus restreint ou à la date de rédaction précise. Ce type de raisonnements a été fréquemment utilisé pour estimer et parfois déterminer la date de rédaction de certaines lettres de la correspondance d'Henri Poincaré.

Dans le cadre de ce travail de représentation de connaissances la granularité est définie au niveau de la journée. En effet, celui-ci permet de considérer les données associées aux lettres de la correspondance pour lesquelles le jour de rédaction est parfois connu. Selon le contexte, ce niveau de granularité pourrait être trop précis ou trop vague - par exemple dans le cas de description de batailles où la granularité devrait être de l'ordre de l'heure voire de la minute. En histoire, il est parfois très difficile d'être précis dans l'estimation de la date associée à un événement. Cela a pour conséquence la définition de dates imprécises qui peuvent être associées à la rédaction d'une lettre, la naissance d'une personne, une nomination au sein d'un établissement, etc. Une telle date est souvent le résultat de choix pris dans l'optique de garantir une cohérence au sein d'une chronologie. Elle correspond à la date hypothétique la plus probable au regard de travaux de recherche.

15. Cet événement constitue l'un des actes fondateurs de l'histoire des applications de l'électricité. Il se déroula à Paris, au Palais de l'Industrie, entre le 10 août 1881 et le 20 novembre 1881 et attira environ 900 000 visiteurs (CARRÉ, 1989). Plusieurs avancées technologiques y furent présentées telles que les ampoules électriques de Thomas Edison, le tramway électrique de Werner von Siemens ou encore le téléphone d'Alexander Graham Bell. Le premier congrès international des électriciens a également été organisé en parallèle de l'exposition.

3 Approches de représentation de connaissances temporelles

Cette section vise à présenter une synthèse des approches de représentation de connaissances temporelles pour les données RDF. Plusieurs ontologies temporelles sont également introduites.

3.1 Associer un fait à une temporalité en RDF

3.1.1 Graphes temporels et réification RDF

Temporal RDF (GUTIERREZ, C. HURTADO et al., 2005) correspond à l'une des premières approches de représentation de données temporelles au sein d'une base RDF. Un graphe temporel est composé d'un ensemble de triplets temporel, triplets étiquetés de la forme $\langle s \ p \ o \rangle : [t]$ où t correspond à la composante temporelle définie comme le moment durant lequel le fait associé au triplet est valide. L'expression $\langle s \ p \ o \rangle : [t_1, t_2]$ est une notation pour $\langle s \ p \ o \rangle : [t \mid t_1 \leq t \leq t_2]$ En pratique, une syntaxe inspirée de la *réification* RDF est introduite pour associer l'étiquette à un triplet au sein d'un graphe RDF classique.

Le concept de réification RDF a été introduit par le W3C pour ajouter des méta-propriétés aux faits représentés par des triplets RDF (MANOLA, MILLER et al., 2004). Pour décrire un triplet, une instance de la classe `rdf:statement` est créée. Les 3 propriétés (`rdf:subject`, `rdf:predicate` et `rdf:object`) sont utilisées pour lier cette instance au triplet existant. C'est cette nouvelle instance qui pourra être décrite par des propriétés additionnelles. Celles-ci peuvent ajouter des informations diverses : un degré de certitude, le ou les auteurs, la provenance, etc. Dans le cas actuel, des méta-triplets peuvent être introduits pour ajouter une composante temporelle aux faits.

L'approche *Temporal* RDF introduit un vocabulaire composé des propriétés `temporal`, `instant`, `interval`, `initial` et `final` et du type littéral `now`. Des règles d'inférence utilisant ces propriétés sont définies et complètent les règles d'inférence RDFS. Celles-ci permettent de passer d'une représentation temporelle à points à une représentation à intervalles et inversement. Un langage d'interrogation de graphes temporels est également introduit bien qu'aucune implémentation ne soit proposée.

Cette approche est étendue par l'ajout de temporalités anonymes afin de représenter et raisonner avec des faits pour lesquels la composante temporelle est inconnue ou incomplète mais néanmoins essentielle (GUTIERREZ, C. A. HURTADO et al., 2006). Ce modèle présente un avantage lié à l'utilisation de méta-propriétés : il est possible d'accéder à un graphe RDF de deux façons différentes selon que l'intérêt soit porté sur la temporalité ou uniquement sur les faits du domaine.

3.1.2 Alternative à la réification standard

Les relations n -aires (*n-ary* relations (NOY, RECTOR et al., 2006)) forment une approche du W3C pour décrire des faits avec des méta-propriétés. Dans le cas

du problème de la temporalité, l’approche *n-ary* consiste à décrire une relation en ajoutant une entité intermédiaire qui sera liée à la « valeur » de la propriété de la relation initiale ainsi qu’à une ou plusieurs composantes temporelles.

RDF* et SPARQL* sont introduits par HARTIG (2017) et forment une alternative à la réification standard. La notion d’imbrication de triplets est introduite et permet de définir un triplet à la position de sujet ou d’objet d’un autre triplet. Cette approche nécessite une extension de la sémantique de RDF et une extension du langage SPARQL. Une fonction permettant de passer d’un graphe RDF* à un graphe RDF classique est fournie. Après transformation, un graphe RDF* sera similaire à un graphe RDF utilisant la réification RDF.

L’utilisation de graphes nommés (*Named graphs*) (CARROLL, BIZER et al., 2005) permet de décomposer un graphe \mathcal{G} en un ensemble de graphes : $\{(\mathcal{G}_1, u_1), (\mathcal{G}_2, u_2), \dots, (\mathcal{G}_n, u_n)\}$ où u_i correspond à une IRI identifiant le graphe \mathcal{G}_i . Un tel graphe peut être décrit avec des méta-propriétés. Un langage, appelé τ -SPARQL, est introduit pour simplifier l’écriture de requêtes SPARQL lorsque l’on souhaite récupérer des méta-données associées à un graphe nommé. L’utilisation de graphes nommés est particulièrement utile lorsque plusieurs faits représentés par des triplets sont associés à la même temporalité mais il n’est pas pertinent dans le cas où la plupart des triplets doit être associé à une temporalité différente. En effet, dans ce cas, il serait nécessaire de créer des graphes nommés composés d’un unique élément.

RDF⁺ (SCHUELER, SIZOV et al., 2008) est une extension de RDF proposée pour la représentation de méta-données s’appuyant sur l’utilisation de graphes nommés. Un fait RDF⁺ correspond à un quintuplet $\langle g \ s \ p \ o \ \theta \rangle$ où g correspond à l’IRI d’un graphe nommé et θ est l’identifiant du fait, avec $\theta \in \Theta$ où l’ensemble Θ est disjoint de l’ensemble des IRI et de l’ensemble des littéraux. L’identifiant θ peut être défini comme le sujet d’un triplet RDF classique afin de décrire le fait initial. Afin d’interroger des graphes RDF⁺, le langage SPARQL est étendu par l’ajout du mot-clé WITH META.

3.1.3 Représentations fondées sur des entités perdurantes

4D fluent (WELTY et FIKES, 2006) est une approche dans laquelle les entités sont décomposées en un ensemble de parties temporelles. Pour ce faire, les notions d’endurants et perdurants sont introduites. Un endurant correspond à une entité qui peut être perçue comme un tout indépendamment d’une quelconque notion temporelle. Un perdurant correspond à un événement ou une entité qui existe partiellement pour chaque instant temporel (cette nouvelle composante temporelle correspond à une 4^e dimension). La proposition de l’approche *4D fluent* est de considérer toutes les entités comme perdurantes, et d’y associer des parties temporelles (*time slices*). Les relations entre des ressources sont définies par des relations entre leurs parties temporelles. Une entité temporelle peut être utilisée pour plusieurs relations si la ressource correspondante est associée à plusieurs faits avec une même temporalité. D’un point de vue ontologique, cette approche peut paraître plus intéressante que d’autres car elle intègre une

sémantique claire : les objets supplémentaires sont des composantes temporelles d’entités perdurantes.

(KRIEGER, KIEFER et al., 2008) proposent de simplifier la représentation fondée sur une vue perdurante. Contrairement à l’approche *4D fluents*, l’entité originale est directement interprétée comme une partie temporelle. Ceci est fait dans le but de simplifier le passage à une représentation temporelle en évitant le besoin de réécrire l’ontologie.

3.1.4 Représentations centrées sur les propriétés

L’approche *singleton property* (NGUYEN, BODENREIDER et SHETH, 2014) introduit un nouveau type de propriétés, appelé *singleton property*, utilisées une unique fois pour un contexte particulier. Une *singleton property* peut être liée à une propriété générique grâce à la propriété `rdf:singletonPropertyOf`. Ainsi, plutôt que d’ajouter des méta-données pour un triplet, un groupe de triplets, ou une ressource, cette méthode permet de décrire la propriété singleton utilisée. Des travaux étendent cette approche en définissant `rdf:singletonPropertyOf` comme une sous propriété de `rdfs:subPropertyOf` (NGUYEN, BODENREIDER, THIRUNARAYAN et al., 2015). Cela a deux conséquences sur les raisonnements : un moteur d’inférences peut désormais reconnaître les triplets utilisant une *singleton property* et il peut également inférer un triplet décrit en utilisant une *singleton property*.

3.1.5 Représentations utilisant un mécanisme d’annotation

D’autres formalismes de représentation s’appuient sur un mécanisme d’annotation. *aRDF* (UDREA, RECUPERO et al., 2010) propose une extension du langage RDF afin de représenter des méta-données associées aux faits. Un fait *aRDF* est décrit par un triplet RDF classique auquel est liée une annotation $a \in \mathcal{A}$. Un ensemble de triplets *aRDF* est exprimé sous la forme $\langle s, p, a, v \rangle$. Différents ensembles peuvent être considérés pour les annotations. Plusieurs exemples sont fournis pour les données temporelles : l’ensemble $\mathcal{A}_{time} = \mathbb{N}$ composé de tous les entiers naturels ordonnés selon la relation « plus petit ou égal ». Cet ensemble représente des instants temporels. Une autre possibilité serait de considérer l’ensemble des intervalles $\mathcal{A}_{time-int} = \{[x, y] \mid x, y \in \mathbb{N}\}$. Un langage de requêtes proposant les mêmes fonctionnalités que le langage SPARQL est introduit pour interroger les graphes *aRDF*.

Le modèle RDFt (ZHANG, WANG et al., 2019) est un modèle permettant l’annotation d’un triplet afin d’y ajouter une composante temporelle. Contrairement aux approches précédentes, ce modèle permet de représenter la validité temporelle d’un fait en conservant le nombre de modifications apportées dans la base.

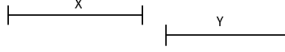
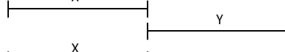
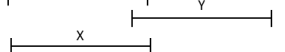
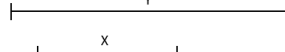
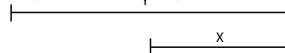
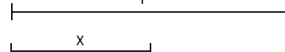
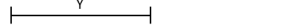
| Relation | Inverse | |
|--------------|-------------------|--|
| X before Y | Y after X |  |
| X meets Y | Y met-by X |  |
| X overlaps Y | Y overlapped-by X |  |
| X starts Y | Y started-by X |  |
| X during Y | Y contains X |  |
| X finishes Y | Y finished-by X |  |
| X equals Y | Y equals X |  |

FIGURE 2 – Algèbre des intervalles d’Allen.

3.2 Ontologies temporelles

Les approches présentées dans cette section sont associées à différentes ontologies pour représenter les éléments temporels. L’objectif de cette partie est de fournir un bref aperçu des principales ontologies temporelles en RDFS et OWL. Pour comparer ces approches, il est notamment pertinent de s’intéresser à l’intégration de l’algèbre des intervalles d’Allen (ALLEN, 1983). Celle-ci définit 13 relations, dont 6 paires de relations inverses (**before** et **after**, **meets** et **met-by**, **overlaps** et **overlapped-by**, **starts** et **started-by**, **finishes** et **finished-by**) et la relation d’égalité **equal** qui est son propre inverse. Ces relations sont détaillées dans la figure 2, qui les explicite en comparant les intervalles temporels X et Y . Cette représentation peut être utilisée pour formaliser des faits et mener des raisonnements. Soient A , B et C 3 intervalles temporels et r_1 et r_2 deux relations telles que $\langle A \ r_1 \ B \rangle$ et $\langle B \ r_2 \ C \rangle$. Selon la nature de r_1 et r_2 , il est possible de déduire ou restreindre la nature de la relation entre A et C . Par exemple, $\langle A \ \text{before} \ B \rangle \wedge \langle B \ \text{before} \ C \rangle \vdash \langle A \ \text{before} \ C \rangle$. Une table de composition est fournie pour l’ensemble des couples de relations (r_1, r_2)

OWL-Time (COX, LITTLE et al., 2020) est l’ontologie recommandée par le W3C pour représenter des éléments temporels au sein d’une base de connaissances RDF. Celle-ci a été introduite en 2006 et la version courante est compatible avec la deuxième version de OWL. Ce modèle introduit plusieurs concepts : les entités temporelles *Instant* et *Interval*, des éléments liés à des systèmes temporels, des éléments pour représenter des positions temporelles et des durées. Les types de données *generalDay*, *generalMonth*, *generalYear* sont proposés comme une extension des types `xsd:gDay`, `xsd:gMonth` et `xsd:gYear` pour représenter des dates dans des calendriers autres que le calendrier grégorien. Plusieurs ressources sont également définies dans cette ontologie afin de représenter les jours de la

semaine, les mois et des unités temporelles. L'ontologie intègre la représentation des 13 relations de l'algèbre des intervalles d'Allen.

Time-determined Web Ontology Language (tOWL (MILEA, FRASINCAR et al., 2011)) est une alternative à OWL-Time correspondant à une extension temporelle du fragment *SR_OIQ(D)* de la logique de descriptions. Cette extension se décompose en 3 couches : l'utilisation de domaines concrets pour représenter des restrictions en utilisant des prédicats binaires, l'introduction de concepts (instants, intervalles) et de relations en incluant la représentation de l'algèbre d'Allen, et une dernière couche proposant des éléments pour une vue perdurante. Cette dernière couche est inspirée de l'approche 4D fluents (WELTY et FIKES, 2006).

SOWL (BATAKIS et PETRAKIS, 2010) est un modèle permettant des représentations et des raisonnements sur les données temporelles et spatiales. Les concepts temporels sont définis grâce à OWL-Time. Le raisonnement est effectué grâce à des règles SWRL. Une extension de SPARQL est également proposée pour intégrer des opérateurs temporels.

HUTO (DIALLO, CORBY et al., 2015) est une ontologie formalisée en RDFS qui permet d'annoter des ressources à l'aide d'expressions temporelles du langage courant. Elle permet de représenter des connaissances temporelles liées à un contexte. 5 types d'expressions temporelles distinctes sont considérées et peuvent ainsi être représentées : des expressions explicites (e.g. 31 décembre 2020, hiver 2020, etc.), des expressions déictiques (e.g. hier, dans 10 jours, etc.), des durées (e.g. 1 heure, 30 jours, etc.), des expressions cycliques (e.g. tous les deux jours, chaque dimanche, etc.) ainsi que des expressions mixtes (e.g. durant une semaine il y a 3 ans). Des règles sont introduites afin d'appliquer des raisonnements sur ces types d'expressions. Parmi les 13 relations issues de l'algèbre des intervalles d'Allen, seules les relations *before* et *after* sont représentées. Contrairement aux autres approches présentées, HUTO intègre la représentation d'éléments dépendant d'un discours.

3.3 Synthèse

Le tableau 1 compare les approches existantes pour la représentation de données temporelles au sein de bases RDF. Plusieurs critères sont utilisés :

Année : année de publication du premier article présentant l'approche.

Extension RDF : une extension de RDF est-elle nécessaire ?

Structure : niveau de structure associé à la représentation (triplet, quadruplet, quintuplet, etc.).

Nombre n-uplets : nombre de n-uplets nécessaires pour représenter un fait en intégrant une composante temporelle.

Sémantique : niveau de sémantique (RDF, RDFS, OWL, etc.).

Langage requêtes : langage de requêtes permettant d'interroger des bases de données pour cette approche.

Ontologie : ontologie associée à la représentation.

| Approche | Année | Extension RDF | Structure | Nombre n-uplets | Sémantique | Langage requêtes | Ontologie |
|--------------------|-------|---------------|-----------------|-----------------|------------------------|---------------------|-----------|
| Temporal RDF | 2005 | Oui | Triplet | 8 | Temporal RDF | Modèle | Modèle |
| Named graphs | 2005 | Non | Quadruplet | 8 | RDFS | τ SPARQL | Libre |
| n-ary | 2006 | Non | Triplet | 5 | OWL | SPARQL | Libre |
| 4D | 2006 | Non | Triplet | 7 | OWL | SPARQL | OWL-Time |
| fluents | | | | | | ext. | |
| RDF ⁺ | 2008 | Oui | Quintuplet | 5 | RDF ⁺ | SPARQL | Libre |
| <i>a</i> RDF | 2010 | Oui | Triplet + Label | N/A | <i>Annotated Logic</i> | Algorithm | Libre |
| Singleton property | 2012 | Oui | Triplet | 5 | ext. RDFS | SPARQL | Libre |
| RDF [*] | 2017 | Oui | Triplet | 7 | RDFS | SPARQL [*] | Libre |
| RDft | 2019 | Oui | Triplet + Label | 8 | ext. RDFS | SPARQL[t] | RDft |

TABLE 1 – Comparaison de modèles de représentation de données temporelles.

Le tableau 2 compare différentes ontologies pour modéliser des éléments temporels. Plusieurs critères de comparaison sont utilisés :

Année : année de publication du premier article présentant l’ontologie.

Reco. W3C : l’ontologie fait-elle l’objet d’une recommandation de la part du W3C ?

Langage : langage de représentation de données associé à l’ontologie.

Intervalles d’Allen : les relations issues de l’algèbre des intervalles d’Allen représentées dans l’ontologie.

Expressions déictiques : l’ontologie intègre-t-elle la représentation d’éléments dépendants d’un discours ?

Granularité : niveaux de granularité permis par l’ontologie.

| Approche | Année | Reco. W3C | Langage | Intervalles d’Allen | Expressions déictiques | Granularité |
|---------------|-------|-----------|-----------|------------------------|------------------------|----------------------|
| Reusable time | 2002 | Non | KIF, OKBC | Toutes | Non | Année à seconde |
| OWL-Time | 2006 | Oui | OWL | Toutes | Non | Année à seconde |
| tOWL | 2010 | Non | OWL | Toutes | Non | Année à seconde |
| sOWL | 2010 | Non | OWL | Toutes | Non | Année à milliseconde |
| SWRL temporal | 2011 | Non | OWL | Toutes | Non | Année à milliseconde |
| HUTO | 2015 | Non | RDFS | <i>before et after</i> | Oui | Siècle à seconde |

TABLE 2 – Comparaison d’ontologies pour la représentation d’éléments temporels.

4 Mise en œuvre et comparaison d’approches pour le corpus d’Henri Poincaré

4.1 Méthodologie

Cette section vise à détailler la mise en œuvre de 4 des approches introduites dans la section précédente : *Temporal* RDF, *4D fluents*, *singleton property* et *n-ary*. Seuls les modèles de représentation ne nécessitant pas d’extension majeure à la sémantique de RDF ont été retenus. Pour chacune des approches sélectionnées, plusieurs aspects seront étudiés et présentés :

- La représentation d’un fait sous la forme de triplets RDF ;
- Une visualisation graphique de cette représentation ;
- La mise en œuvre de trois requêtes SPARQL.

Le fait à représenter est lié à la vie d’Henri Poincaré :

$\mathcal{F} =$ « Henri Poincaré a vécu à Paris de 1881 à 1912. »

Dans la base actuelle, ce fait est représenté sous la forme d’une relation binaire `livesAt(henriPoincaré, Paris)`. Henri Poincaré correspond à une ressource décrite dans la base du corpus de la correspondance¹⁶ tandis que Paris est définie comme une ressource *Geonames*¹⁷. Comme expliqué dans la section 2, cette représentation n’est pas adaptée et il est nécessaire d’y ajouter une composante temporelle. Chacune des approches de représentation sera implémentée avec l’ontologie OWL-Time (sauf *Temporal* RDF qui introduit un vocabulaire temporel).

Voici les trois requêtes qui devront être formulées et qui sont ici exprimées de façon informelle :

$\mathcal{Q}_1 =$ « Donner le lieu de résidence d’Henri Poincaré en 1900. »

$\mathcal{Q}_2 =$ « Donner les personnes qui ont enseigné dans un même établissement qu’Henri Poincaré entre 1894 et 1906. »

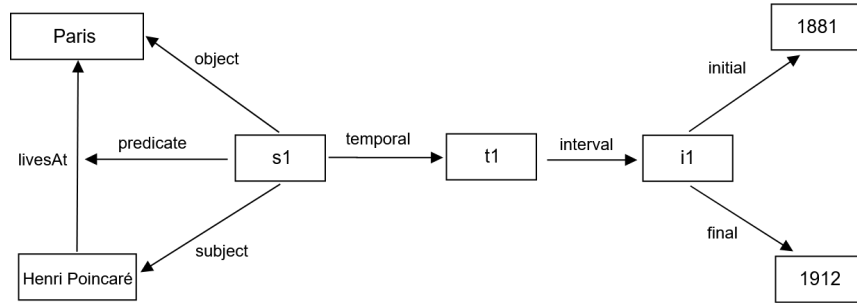
$\mathcal{Q}_3 =$ « Donner les périodes temporelles durant lesquelles Henri Poincaré était président du Bureau des longitudes. »

L’objectif de \mathcal{Q}_1 est de retourner la ressource Geonames correspondant au lieu de résidence d’Henri Poincaré en 1900. \mathcal{Q}_2 correspond à la recherche de personnes qui auraient pu côtoyer Poincaré (de par leurs responsabilités pédagogiques) durant l’intervalle temporel [1894, 1906] qui correspond à la période durant laquelle *l’affaire Dreyfus*¹⁸ divisait la France, du moment où le scandale éclatait

16. <http://henripoincare.fr/api/items/843>.

17. <https://www.geonames.org/2988507>.

18. L’affaire Dreyfus correspond à un conflit social et politique qui a bouleversé la France à la fin du XIX^e siècle et au début du XX^e siècle. Celui-ci fait suite à l’accusation de trahison faite au capitaine Alfred Dreyfus. Plusieurs personnalités s’impliquent dans cette affaire en rejoignant l’un des deux camps en opposition : d’un côté les « dreyfusards », qui soutiennent le capitaine et prône son innocence et de l’autre, les « antidreyfusards », convaincus de sa culpabilité. Cette affaire connaît un rebond suite à la publication d’un article d’Émile Zola, *J’accuse... !*, où celui-ci, scandalisé par le verdict, prend la défense de Dreyfus et lance des accusations nominatives à l’encontre de dix acteurs de l’affaire. Pour plus d’informations concernant l’implication d’Henri Poincaré dans cette affaire, voir (ROLLET, 1997).



(a) Représentation graphique de \mathcal{F} avec l'approche *Temporal* RDF.

```

(henriPoincaré livesAt paris)
(s1 subject henriPoincaré)
(s1 predicate livesAt)
(s1 object paris)
(s1 temporal t1)
(t1 interval i1)
(i1 initial 1881)
(i1 final 1912)

```

(b) Triplets représentant \mathcal{F} avec l'approche *Temporal* RDF.

FIGURE 3 – Représentation de \mathcal{F} avec l'approche *Temporal* RDF.

jusqu'à la réhabilitation du capitaine Dreyfus¹⁹. L'idée de cette requête est d'identifier des personnalités liées au corpus qui auraient pu discuter de cette affaire avec Henri Poincaré. \mathcal{Q}_3 doit retourner une réunion d'intervalles disjoints car Henri Poincaré a été président du Bureau des longitudes de janvier 1899 à janvier 1900 et une seconde fois entre février 1909 et décembre 1910.

4.2 Temporal RDF

Pour ajouter une temporalité à un fait, cette méthode propose une syntaxe fondée sur la réification RDF. Une ontologie est également associée à la représentation proposée. Pour représenter \mathcal{F} , une instance de la classe *statement* est créée et celle-ci est liée au triplet `<henriPoincaré livesAt paris>` grâce aux propriétés `subject`, `predicate` et `object`. Ce *statement* est défini par une temporalité `t1` associée à un intervalle temporel `i1`. `i1` est décrit par une date de début (1881) et une date de fin (1912). La figure 3 présente une visualisation graphique de cette représentation. Cette méthode de représentation porte le total de triplets à 8 comme l'illustre la figure 3.

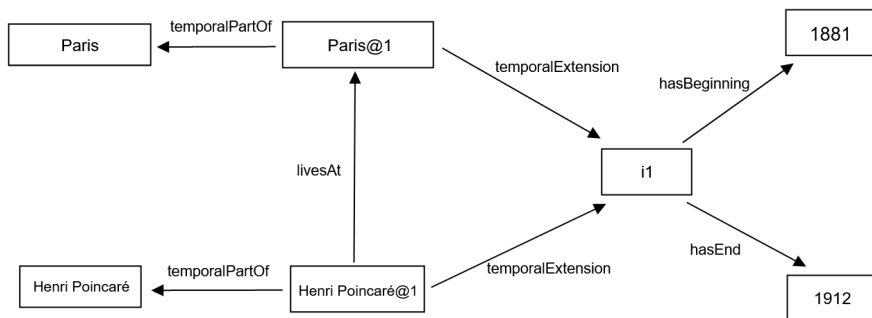
¹⁹. Cet intervalle stricte est choisi pour l'exemple de la requête. Dans le cadre de recherches en histoire, il est souvent nécessaire de considérer des intervalles définis de façon moins précise. Ce point est abordé dans la suite de l'article.

La formulation de requêtes SPARQL est ici assez lourde car il est nécessaire d'identifier le *statement* afin d'accéder à la temporalité, ce qui augmente le nombre de patrons de triplets nécessaires. Ce problème est illustré par Q_1^{TR} , Q_2^{TR} et Q_3^{TR} , requêtes formelles associées aux requêtes Q_1 , Q_2 et Q_3 . Cependant, un avantage est que le triplet original n'est pas modifié et il est simple d'accéder à une donnée sans s'intéresser à la temporalité. Par exemple, pour retrouver l'ensemble des lieux de résidence de Poincaré qui sont définis dans la base, la requête à formuler est peu complexe et reste la même qu'il y ait ou non une temporalité associée aux faits.

| | |
|---------------------|--|
| $Q_1^{\text{TR}} =$ | <pre> SELECT ?loc WHERE {?s1 subject henriPoincaré . ?s1 predicate livesAt . ?s1 object ?loc . ?s1 temporal ?t1 . ?t1 interval ?i1 . ?i1 initial ?d1 . ?i1 final ?d2 . FILTER(?d1 <= 1900 AND ?d2 >= 1900)} </pre> |
| $Q_2^{\text{TR}} =$ | <pre> SELECT ?pers WHERE {?s1 subject henriPoincaré . ?s1 predicate teachAt . ?s1 object ?loc . ?s2 subject ?pers . ?s2 predicate teachAt . ?s2 object ?loc . ?s2 temporal ?t1 . ?t1 interval ?i1 . ?i1 initial ?d1 . ?i1 final ?d2 . ?s2 temporal ?t2 . ?t2 interval ?i2 . ?i2 initial ?d3 . ?i2 final ?d4 . FILTER(?d1 >= 1894 AND ?d2 <= 1906 AND ?d3 >= 1894 AND ?d4 <= 1906)} </pre> |
| $Q_3^{\text{TR}} =$ | <pre> SELECT ?d1 ?d2 WHERE {?s1 subject henriPoincaré . ?s1 predicate presidentOf . ?s1 object bureauDesLongitudes . ?s1 temporal ?t1 . ?t1 interval ?i1 . ?i1 initial ?d1 . ?i1 final ?d2} </pre> |

4.3 4D fluents

Cette approche propose de décomposer les ressources en un ensemble de parties temporelles. Afin de représenter \mathcal{F} , il est nécessaire d'ajouter une partie temporelle aux ressources identifiant Henri Poincaré et Paris. Ces parties temporelles sont toutes les deux liées à une temporalité correspondant à un intervalle. Une date de début (1881) et de fin (1912) sont associées à cet intervalle temporel. Au total, 7 triplets sont nécessaires pour représenter \mathcal{F} comme l'illustre la figure 4. Les requêtes Q_1^{4D} , Q_2^{4D} et Q_3^{4D} mettent en avant la complexité des



(a) Représentation graphique de \mathcal{F} avec l'approche $4D$ fluents.

```

⟨henriPoincaré@1 temporalPartOf henriPoincaré⟩
⟨paris@1 temporalPartOf paris⟩
⟨henriPoincaré@1 livesAt paris@1⟩
⟨henriPoincaré@1 temporalExtension i1⟩
⟨paris@1 temporalExtension i1⟩
⟨i1 hasBeginning 1881⟩
⟨i1 hasEnd 1912⟩

```

(b) Triplets représentant \mathcal{F} avec l'approche $4D$ fluents.

FIGURE 4 – Représentation de \mathcal{F} avec l'approche $4D$ fluents.

requêtes SPARQL pour l'approche *4D fluents*. Il est, pour chacune de ces requêtes, nécessaire de préciser les liens entre les ressources, les parties temporelles et les temporalités.

$$Q_1^{4D} = \left| \begin{array}{l} \text{SELECT ?loc} \\ \text{WHERE \{?p1 temporalPartOf henriPoincaré .} \\ \quad \text{?p2 temporalPartOf ?loc . ?p1 livesAt ?p2 .} \\ \quad \text{?p1 temporalExtension ?i1 .} \\ \quad \text{?p2 temporalExtension ?i1 .} \\ \quad \text{?i1 hasBeginning ?d1 . ?i1 hasEnd ?d2 .} \\ \text{FILTER(?d1 <= 1900 AND ?d2 >= 1900)\}} \end{array} \right.$$

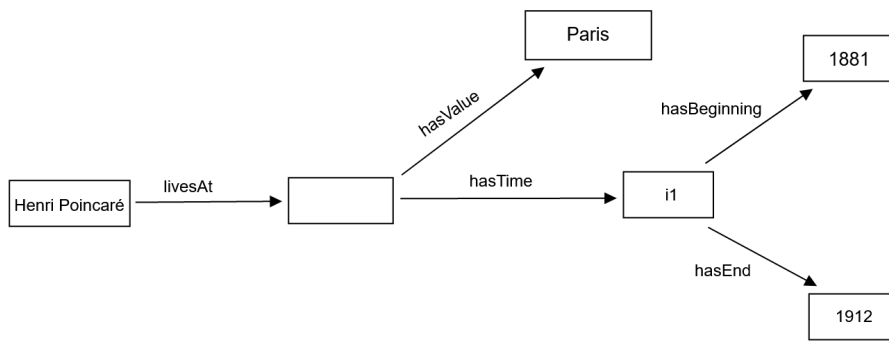
$$Q_2^{4D} = \left| \begin{array}{l} \text{SELECT ?person} \\ \text{WHERE \{?p1 temporalPartOf henriPoincaré .} \\ \quad \text{?p2 temporalPartOf teachAt .} \\ \quad \text{?p3 temporalPartOf ?person .} \\ \quad \text{?p4 temporalPartOf teachAt .} \\ \quad \text{?p1 temporalExtension ?i1} \\ \quad \text{?p2 temporalExtension ?i1 .} \\ \quad \text{?p3 temporalExtension ?i2} \\ \quad \text{?p4 temporalExtension ?i2 .} \\ \quad \text{?i1 hasBeginning ?d1 . ?i1 hasEnd ?d2 .} \\ \quad \text{?i2 hasBeginning ?d3 . ?i2 hasEnd ?d4 .} \\ \text{FILTER(?d1 >= 1894 AND ?d2 <= 1906} \\ \quad \text{AND ?d3 >= 1894 AND ?d4 <= 1906)\}} \end{array} \right.$$

$$Q_3^{4D} = \left| \begin{array}{l} \text{SELECT ?d1 ?d2} \\ \text{WHERE \{?p1 temporalPartOf henriPoincaré .} \\ \quad \text{?p2 temporalPartOf bureauDesLongitudes .} \\ \quad \text{?p1 presidentOf ?p2 .} \\ \quad \text{?p1 temporalExtension ?i1 .} \\ \quad \text{?p2 temporalExtension ?i1 .} \\ \quad \text{?i1 hasBeginning ?d1 . ?i1 hasEnd ?d2}\}} \end{array} \right.$$

4.4 Singleton property

Dans le cas de l'approche *singleton property*, peu de triplets sont nécessaires pour décrire un fait en incluant une temporalité. La propriété `livesAt#1` est définie comme une `singletonPropertyOf` de la propriété `livesAt`. Il est ainsi nécessaire de créer une propriété `livesAt#i` à chaque fois que l'on souhaite utiliser la propriété `livesAt` en précisant une temporalité. \mathcal{F} est représenté par 5 triplets comme illustré dans la figure ??.

L'introduction de *singleton propriétés* ne complexifie que peu l'écriture de requêtes SPARQL. Les requêtes Q_1^{SP} , Q_2^{SP} et Q_3^{SP} comportent un faible nombre de patrons de triplets par rapport aux deux approches précédentes. De plus, l'extension proposée dans (NGUYEN, BODENREIDER, THIRUNARAYAN et al., 2015)



(a) Représentation graphique de \mathcal{F} avec l'approche n -ary.

```

⟨henriPoincaré livesAt ?node⟩
⟨?node hasValue paris⟩
⟨?node hasTime i1⟩
⟨i1 hasBeginning 1881⟩
⟨i1 hasEnd 1912⟩
  
```

(b) Triplets représentant \mathcal{F} avec l'approche n -ary.

FIGURE 5 – Représentation de \mathcal{F} avec l'approche n -ary.

permet de simplifier la formulation de requêtes pour lesquelles la temporalité d'un fait ne doit pas être retournée.

$$Q_1^{\text{SP}} = \left\{ \begin{array}{l} \text{SELECT ?loc} \\ \text{WHERE \{henriPoincaré ?p ?loc .} \\ \quad \text{?p singletonPropertyOf livesAt .} \\ \quad \text{?p hasTime ?i1 .} \\ \quad \text{?i1 hasBeginning ?d1 . ?i1 hasEnd ?d2 .} \\ \text{FILTER(?d1 <= 1900 AND ?d2 >= 1900)\}} \end{array} \right.$$

$$Q_2^{\text{SP}} = \left\{ \begin{array}{l} \text{SELECT ?person} \\ \text{WHERE \{henriPoincaré ?p1 ?loc .} \\ \quad \text{?person ?p2 ?loc .} \\ \quad \text{?p1 singletonPropertyOf teachAt .} \\ \quad \text{?p2 singletonPropertyOf teachAt .} \\ \quad \text{?p1 hasTime ?i1 . ?p2 hasTime ?i2 .} \\ \quad \text{?i1 hasBeginning ?d1 . ?i1 hasEnd ?d2 .} \\ \quad \text{?i2 hasBeginning ?d3 . ?i2 hasEnd ?d4 .} \\ \text{FILTER(?d1 >= 1894 AND ?d2 <= 1906} \\ \quad \text{AND ?d3 >= 1894 AND ?d4 <= 1906)\}} \end{array} \right.$$

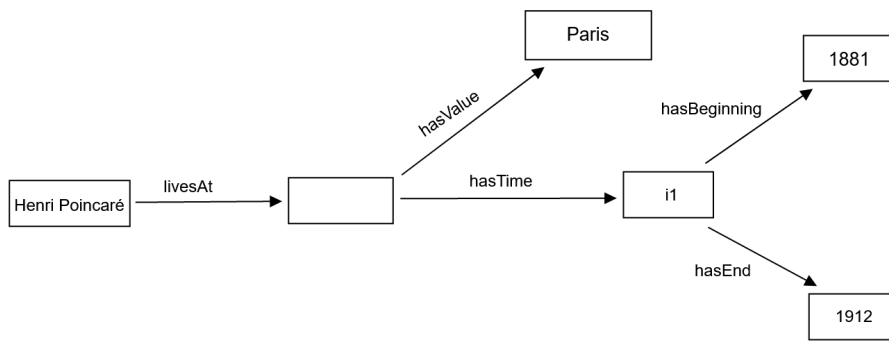
$$Q_3^{\text{SP}} = \left\{ \begin{array}{l} \text{SELECT ?d1 ?d2} \\ \text{WHERE \{henriPoincaré ?p bureauDesLongitudes .} \\ \quad \text{?p singletonPropertyOf presidentOf .} \\ \quad \text{?p hasTime ?i1} \\ \quad \text{?i1 hasBeginning ?d1 . ?i1 hasEnd ?d2\}} \end{array} \right.$$

4.5 n-ary

Cette approche représente \mathcal{F} en simulant une relation ternaire grâce à l'introduction d'un nœud vide. Ce nœud permet de lier l'utilisation de la propriété `livesAt` à une valeur et à une temporalité. La figure 6 illustre cette représentation où 5 triplets sont nécessaires.

Comme dans le cas de l'approche *singleton property* les requêtes $Q_1^{n\text{-ary}}$, $Q_2^{n\text{-ary}}$ et $Q_3^{n\text{-ary}}$ comportent un faible nombre de patrons de triplets. L'approche *n-ary* impose de travailler avec des nœuds vides mais ce mécanisme s'intègre bien aux requêtes SPARQL. Cependant, l'assertion du triplet original est perdue, ce qui oblige à ajouter un patron de triplet dans les requêtes lorsqu'on ne souhaite pas récupérer la temporalité (par rapport à une version sans représentation de la temporalité).

$$Q_1^{n\text{-ary}} = \left\{ \begin{array}{l} \text{SELECT ?loc} \\ \text{WHERE \{henriPoincaré livesAt ?node .} \\ \quad \text{?node hasValue ?loc . ?node hasTime ?i1 .} \\ \quad \text{?i1 hasBeginning ?d1 . ?i1 hasEnd ?d2 .} \\ \text{FILTER(?d1 <= 1900 AND ?d2 >= 1900)\}} \end{array} \right.$$



(a) Représentation graphique de \mathcal{F} avec l'approche n -ary.

```

⟨henriPoincaré livesAt ?node⟩
⟨?node hasValue paris⟩
⟨?node hasTime i1⟩
⟨i1 hasBeginning 1881⟩
⟨i1 hasEnd 1912⟩

```

(b) Triplets représentant \mathcal{F} avec l'approche n -ary.

FIGURE 6 – Représentation de \mathcal{F} avec l'approche n -ary.

| | |
|------------------------|---|
| $Q_2^{n\text{-ary}} =$ | <pre> SELECT ?person WHERE {henriPoincaré teachAt ?node1 . ?node1 hasValue ?loc ?node1 hasTime ?i1 . ?person teachAt ?node2 . ?node2 hasValue ?loc . ?node2 hasTime ?i2 . ?i1 hasBeginning ?d1 . ?i1 hasEnd ?d2 . ?i2 hasBeginning ?d3 . ?i2 hasEnd ?d4 . FILTER(?d1 >= 1894 AND ?d2 <= 1906 AND ?d3 >= 1894 AND ?d4 <= 1906)} </pre> |
| $Q_3^{n\text{-ary}} =$ | <pre> SELECT ?d1 ?d2 WHERE {henriPoincaré presidentOf ?node . ?node hasValue bureauDesLongitudes . ?node hasTime ?i1 . ?i1 hasBeginning ?d1 . ?i1 hasEnd ?d2} </pre> |

4.6 Choix de l'approche

La mise en œuvre de ces différentes approches a permis de mettre en avant leurs avantages et inconvénients pour des cas concrets d'édition et de recherche d'informations. Dans le cadre des travaux autour du corpus d'Henri Poincaré, il a finalement été choisi d'opter pour le modèle fondé sur les relations n -aires (n -ary). Cette approche est l'une des plus intuitives et fait l'objet d'une recommandation de la part du W3C. Aucune extension à la sémantique de RDF n'est nécessaire et le nombre de triplets supplémentaires introduits par la représentation reste faible par rapport à d'autres modèles. De plus, la structure proposée permet d'ajouter des éléments supplémentaires afin de décrire les faits (degré de certitude, auteur, commentaire, etc.).

La section suivante introduit l'utilisation d'un mécanisme de transformations de requêtes SPARQL pour simplifier l'écriture de requêtes et interroger des graphes temporels structurés par des relations n -aires.

5 Application d'un mécanisme de transformation de requêtes SPARQL

Cette section présente l'outil SQTRL (BRUNEAU, GAILLARD et al., 2017), qui permet la définition et l'application de règles de transformation de requêtes SPARQL. Son utilisation permet ici de répondre à deux problématiques : simplifier l'écriture de requêtes SPARQL lorsqu'est utilisée une représentation de type n -ary et mener des raisonnements avec des données temporelles.

5.1 SPARQL Query Transformation Rule Language

SQTRL est un outil associé à un langage qui a été introduit pour permettre des requêtes flexibles avec le langage SPARQL. Ce mécanisme fonctionne par la

définition et l'application de règles de transformation de requêtes. Une règle peut être appliquée pour générer de nouvelles requêtes à partir d'une requête initiale et d'une base RDF. Les règles peuvent être indépendantes d'une application (généralisation, spécialisation, etc.) ou dépendantes d'une application (échange de l'expéditeur et du destinataire de la lettre, remplacement du thème de la lettre par un thème lié, etc.).

Une règle SCTRL est définie par un ensemble de champs :

- `name(r)` : une chaîne de caractères identifiant la règle ;
- `context(r)` : un patron de triplets RDF à rechercher dans la base RDF ;
- `left(r)` : un patron de triplets RDF présent dans le corps de la requête initiale Q ;
- `right(r)` : un patron de triplets RDF qui remplace `left(r)` afin de générer Q' ;
- `cost(r)` : un entier positif représentant le coût de transformation ;
- `explanation(r)` : un texte décrivant de façon informelle la règle de transformation et pouvant contenir des variables issues de `context`, `left` et `right`.

Soit \mathcal{G} un graphe RDF, Q une requête SPARQL, et r une règle SCTRL, r peut être appliquée pour Q si `context(r)` peut être appareillé à \mathcal{G} et `left(r)` peut être appareillé au corps de Q . L'application de r sur Q permet de générer Q' en substituant `left(r)` par `right(r)`. Un coût est associé à chacune des règles pour privilégier l'application de certaines règles.

Les règles sont représentées en utilisant une syntaxe XML. La figure 7 présente un exemple de règle qui permet de remplacer une instance (`?o`) par une variable (`?x`). Cette variable a pour type l'une des classes à laquelle appartient l'instance (`?C`).

```
<rule name="Généralise instance objet">
  <context>?o a ?C</context>
  <left>?s ?p ?o</left>
  <right>?s ?p ?x . ?x a ?C</right>
  <cost>2.0</cost>
  <explanation>Généralise ?o par une instance de ?C </explanation>
</rule>
```

FIGURE 7 – Un exemple de règle SCTRL définie avec une syntaxe XML.

5.2 Simplifier l'écriture de requêtes SPARQL

La proposition est de réutiliser ce mécanisme pour simplifier l'écriture de requêtes SPARQL. Imaginez un historien à la recherche d'informations à propos des lieux de résidence d'Henri Poincaré. Deux cas de figure sont envisagés : soit l'historien souhaite retrouver l'ensemble des lieux de résidence d'Henri Poincaré sans information temporelle soit il souhaite lister les couples (lieu, intervalle temporel). Ces besoins pourraient mener à la formulation des requêtes Q_{h1} et Q_{h2} . Dans les deux cas, il est nécessaire d'introduire un nœud vide pour retrouver les

valeurs associées à des représentations n -aires. Bien que les nœuds vides soient faciles à intégrer aux requêtes SPARQL, les supprimer permettrait de gagner en lisibilité. Une solution à ce problème serait de laisser à l'utilisateur la possibilité de saisir une requête incomplète qui serait ensuite compilée par un outil vers la requête correcte, qui elle tiendrait compte de la représentation n -aire. Un moyen d'implémenter un tel système serait d'utiliser SQTRL afin de définir des règles de transformations syntaxiques.

$$Q_{h1} = \left| \begin{array}{l} \text{SELECT ?loc} \\ \text{WHERE \{henriPoincaré livesAt ?node .} \\ \quad \text{?node hasValue ?loc .} \\ \quad \text{\}} \end{array} \right.$$

$$Q_{h2} = \left| \begin{array}{l} \text{SELECT ?loc ?d1 ?d2} \\ \text{WHERE \{henriPoincaré livesAt ?node .} \\ \quad \text{?node hasValue ?loc .} \\ \quad \text{?node hasTime ?i1 .} \\ \quad \text{?i1 hasBeginning ?d1 .} \\ \quad \text{?i1 hasEnd ?d2} \\ \quad \text{\}} \end{array} \right.$$

Pour cela, les règles r_1 et r_2 sont introduites (figures 8 et 9). La première permet de récupérer la valeur d'une propriété et la deuxième permet de récupérer la valeur avec la temporalité associée. Dans les deux cas, un utilisateur n'a pas besoin de tenir compte de la structure associée à la représentation n -aire. Il peut saisir une requête de la forme `SELECT ?o WHERE {?s ?p ?o}` et le système générera une requête contenant le patron de triplets complet permettant d'accéder à la valeur et, si nécessaire, à la temporalité.

```
<rule name="Récupération de valeur">
  <context>?x ?p ?y . ?y ahp:hasValue ?v</context>
  <left>?s ?p ?o</left>
  <right>?s ?p ?node . ?node ahp:hasValue ?o</right>
  <cost>0</cost>
  <explanation>
    Adapte le patron de triplet pour tenir compte de la représentation
    n-aire et récupérer la valeur associée à la propriété ?p.
  </explanation>
</rule>
```

FIGURE 8 – Règle de transformation r_1 .

Un outil Web implémentant ces règles de transformations a été mis en place. L'utilisateur peut choisir un mode de formulation de requêtes, selon qu'il s'intéresse ou non à la temporalité. Selon le mode sélectionné, l'une des deux règles de transformation s'applique pour retourner les valeurs correspondant à la recherche. La figure 10 illustre l'application de r_2 pour une requête visant à récupérer les lieux de vie d'Henri Poincaré avec la temporalité associée.

```

<rule name="Récupération de valeur et temporalité">
  <context>
    ?x ?p ?y . ?y ahpo:hasValue ?v .
    ?y time:hasTime ?t . ?t time:hasBeginning ?b . ?t time:hasEnd ?e
  </context>
  <left>?s ?p ?o</left>
  <right>
    ?s ?p ?node . ?node ahpo:hasValue ?o
    ?node time:hasTime ?i . ?i time:hasBeginning?d1 . ?i time:hasEnd ?d2
  </right>
  <cost>0</cost>
  <explanation>
    Adapte le patron de triplet pour tenir compte de la représentation
    n-aire et récupérer la valeur et la temporalité associées à ?p.
  </explanation>
</rule>

```

FIGURE 9 – Règle de transformation r_2 .

5.3 Mener des raisonnements avec les données temporelles

Le choix de l'ontologie *OWL-Time*, par l'intégration de l'algèbre des intervalles d'Allen, permet de représenter des relations entre différentes temporalités associées à des faits du corpus d'Henri Poincaré. Dans certains cas, des temporalités sont manquantes, car les sources disponibles n'ont pas permis d'estimer ces éléments temporels. Il advient également qu'une information connue ne soit pas encore représentée dans la base de connaissances. De ce fait, lorsqu'une personne interroge le corpus grâce au langage SPARQL, plusieurs ressources peuvent ne pas être retournées par le système alors qu'elles pourraient apporter des éléments de réponse à une problématique de recherche. Une idée pour pallier à ce problème est de mettre en place des règles d'inférences personnalisées qui s'appuieraient sur des connaissances liées à une ontologie. Pour cela, la proposition est d'utiliser le langage SQTRL, non pas pour représenter des règles de transformation de requêtes SPARQL, mais pour créer des règles de saturation de graphes RDF temporels.

Soit \mathcal{G}_{ex} un graphe tel que :

$$\mathcal{G}_{ex} = \{ \langle \text{letter11 sentBy henriPoincaré} \rangle, \\ \langle \text{letter11 sentTo göstaMittagLeffler} \rangle, \\ \langle \text{letter11 hasWritingDate t1} \rangle, \\ \langle \text{letter22 sentBy henriPoincaré} \rangle, \\ \langle \text{letter22 sentTo göstaMittagLeffler} \rangle, \\ \langle \text{letter22 hasWritingDate t2} \rangle, \\ \langle \text{t2 hasTime 1883} \rangle, \\ \langle \text{letter11 hasReply letter22} \rangle \}$$

Enter your SPARQL query

```
PREFIX ahpo: <http://e-hp.ahp-numerique.fr/ahpo#>
```

```
#Retrieve the living places of Henri Poincaré
```

```
SELECT ?loc
WHERE {
  <http://henripoincare.fr/api/items/843> ahpo:livesAt ?loc
}
```

Options

- With temporal data
- Without temporal data

Limit results

Transform query and get results

Generated SPARQL query

```
PREFIX ahpo: <http://e-hp.ahp-numerique.fr/ahpo#>
```

```
PREFIX time: <http://www.w3.org/2006/time#>
```

```
#Retrieve the living places (with temporal data) of Henri Poincaré
```

```
SELECT ?loc ?d1 ?d2
WHERE {
  <http://henripoincare.fr/api/items/843> ahpo:livesAt ?node .
  ?node ahpo:hasValue ?loc .
  ?node time:hasTime ?i .
  ?i time:hasBeginning ?d1 .
  ?i time:hasEnd ?d2
}
```

FIGURE 10 – Interface associée à l’outil de transformation de requêtes SPARQL.

Imaginons un historien effectuant une recherche sur ce graphe et formulant la requête suivante :

$$Q_{1885} = \begin{array}{l} \text{« Donner les lettres envoyées} \\ \text{avant 1885 par Henri Poincaré à} \\ \text{Gösta Mittag-Leffler. »} \end{array}$$

Le système doit retourner `letter22` car sa date d'expédition est renseignée et correspond à la contrainte définie dans Q_{1885} . En revanche, `letter11` n'est pas retournée car sa temporalité est inconnue. Or, la base contient une information importante : `letter11` a pour réponse `letter22`. Il est donc possible de déduire que la date de rédaction de `letter11` est antérieure à celle de `letter22`, et donc antérieure à 1885. Une manière d'automatiser ce raisonnement est d'explicitier la relation entre les deux dates de rédaction. Pour cela, il est nécessaire d'ajouter le triplet `(t1 before t2)`.

Pour rendre ce type de raisonnements automatique, un nouveau type de règles SQTRL est introduit pour saturer des graphes RDF. Dans le cas de l'exemple, la règle *r_{hasReply}* (figure 11) permet d'ajouter des relations entre les temporalités liées à la date de rédaction de lettres de la correspondance. Ce nouveau type de règles introduit le champ `new` qui remplace les champs `left` et `right`. Ce champ correspond aux triplets RDF devant être ajoutés à la base de connaissances.

```
<rule name="A pour réponse">
  <context>
    ?l1 hasWritingDate ?t1 . ?l2 hasWritingDate ?t2 .
    ?l1 hasReply ?l2
  </context>
  <new>?t1 before ?t2</new>
  <explanation>
    Ajoute une relation entre les temporalités associées à des
    dates de rédaction de lettres.
  </explanation>
</rule>
```

FIGURE 11 – Exemple de règle de saturation de graphe RDF.

Ce mécanisme de saturation de graphes est utile pour expliciter des relations entre des éléments temporels. Il s'appuie sur des raisonnements mis en place par des humains et représentés grâce à des règles SQTRL dont l'application permet l'ajout de triplets pour compléter des bases RDF. Le système implémenté prévient l'application de règles qui engendreraient l'ajout de triplet déjà présents dans la base de connaissances. Cette section a présenté un exemple lié aux dates de rédaction des lettres de la correspondance mais d'autres règles ont été définies pour le corpus de la correspondance d'Henri Poincaré. Par exemple, une règle a pour objectif d'estimer la date de rédaction d'un article scientifique en s'appuyant sur les dates de rédaction de lettres qui le mentionnent. Une autre règle a pour objectif de préciser les intervalles temporels durant lesquels Henri Poincaré et d'autres personnes ont enseigné conjointement au sein d'un même établissement.

6 Conclusion

La représentation des connaissances issues de corpus historiques nécessite l'intégration de composantes temporelles. Pour les corpus indexés à l'aide des technologies du Web sémantique, cette tâche peut s'avérer difficile car le modèle RDF est fondé sur des relations binaires. Différentes structures de représentations des données temporelles ont été proposées, certaines nécessitant une extension des modèles standards du Web sémantique. Pour l'exploitation du corpus lié à la correspondance et aux écrits d'Henri Poincaré, il a été décidé d'utiliser des représentations n -aires. Ce type de représentations permet de répondre aux besoins énoncés dans l'introduction de cet article : décrire un fait en y associant une temporalité (moments et intervalles) ; ne pas introduire d'extension à RDF ; permettre l'utilisation du langage SPARQL pour accéder aux faits sans ou avec la temporalité associée. Dans le cadre de ces travaux, un mécanisme de transformation de requêtes SPARQL a été utilisé pour simplifier la syntaxe de requêtes impliquant des représentations n -aires. Associé à une interface, cet outil permet aux utilisateurs de ne pas tenir compte de la représentation n -aire et de récupérer simplement un fait seul ou un fait avec sa temporalité. Une extension de l'outil SQTRL permet également de saturer des graphes RDF temporels à l'aide de règles personnalisées qui exploitent la sémantique des propriétés de l'ontologie. L'idée est d'explicitier des connaissances qui peuvent être dégagées par l'application de raisonnements.

L'un des avantages dans le choix de représentations n -aires est que cette structure peut également être réutilisée pour décrire un fait à l'aide d'éléments additionnels à la temporalité. Il serait notamment pertinent d'associer un degré de certitude ou un point de vue à certains des faits de la base de connaissances. Intégrer cette part de subjectivité pourrait aider à représenter les connaissances de manière plus fidèle. Un autre travail futur concerne la représentation de connaissances imprécises. Lors de recherches en histoire, il est souvent complexe de définir avec précision la date d'un événement. Dans le cadre du corpus d'Henri Poincaré, cela peut correspondre à la date d'une union, d'une nomination, de rédaction d'une lettre, etc. Un historien doit faire des choix pour garantir la cohérence de récits mais il serait pertinent d'associer à la représentation d'un fait une temporalité floue qui rendrait compte de l'information telle que perçue par l'historien. Des travaux ont été menés pour intégrer la notion de « flou » lors de la représentation des connaissances temporelles. Notamment, une extension de l'ontologie OWL-Time permet de décrire des intervalles temporels incertains à l'aide des propriétés `fuzzyHasBeginning` et `fuzzyHasEnd` (GHORBEL, HAMDI et al., 2018).

Remerciements

Ce travail a bénéficié d'une aide de l'État, gérée par l'Agence Nationale de la Recherche, au titre du projet Investissements d'Avenir Lorraine Université d'Excellence, portant la référence ANR-15-IDEX-04-LUE.

Références

- ALLEN, James F (1983). “Maintaining knowledge about temporal intervals”. *Communications of the ACM* 26.11. Sous la dir. de Peter J. DENNING, p. 832-843.
- ANTONIOU, Grigoris et Frank VAN HARMELEN (2004). “Web ontology language : Owl”. *Handbook on ontologies*. Sous la dir. de Steffen STAAB et Rudi STUDER. Berlin, Heidelberg : Springer Berlin Heidelberg, p. 67-92.
- BATSAKIS, Sotiris et Euripides G. M. PETRAKIS (2010). “SOWL : Spatio-Temporal Representation, Reasoning and Querying over the Semantic Web”. *Proceedings of the 6th International Conference on Semantic Systems*. Sous la dir. d’Adrian PASCHKE. Graz, Austria : Association for Computing Machinery.
- BOULAIRE, Cécile et Romeo CARABELLI (2017). “Du digital naïve au bricoleur numérique : les images et le logiciel Omeka”. *Expérimenter les humanités numériques. Des outils individuels aux projets collectifs*. Sous la dir. de Étienne CAVALIÉ, Frédéric CLAVERT, Olivier LEGENDRE et Dana MARTIN. Montréal, Québec : Les Presses de l’Université de Montréal. Chap. 7, p. 81-103.
- BRICKLEY, D. et R. V. GUHA (2014). *RDF Schema 1.1, W3C recommendation*. <https://www.w3.org/TR/rdf-schema/>. Dernière consultation : août 2020.
- BRUNEAU, Olivier, Emmanuelle GAILLARD, Nicolas LASOLLE, Jean LIEBER, Emmanuel NAUER et Justine REYNAUD (2017). “A SPARQL Query Transformation Rule Language — Application to Retrieval and Adaptation in Case-Based Reasoning”. *Case-Based Reasoning Research and Development. ICCBR 2017*. Sous la dir. de David AHA et Jean LIEBER. Lecture Notes in Computer Science. Cham : Springer, p. 76-91.
- CARLYLE, Thomas (1830). “On History”. *Thomas Carlyle : Selected Writings*. Sous la dir. d’Alan SHELSTON. Penguin Books, London, p. 51-58.
- CAROTHERS, Gavin et Eric PRUD’HOMMEAUX (fév. 2014). *RDF 1.1 Turtle*. <http://www.w3.org/TR/2014/REC-turtle-20140225/>. W3C Recommendation. Dernière consultation : novembre 2020.
- CARRÉ, Patrice A (1989). “Expositions et modernité : Electricité et communication dans les expositions parisiennes de 1867 à 1900”. *Romantisme* 19.65, p. 37-48.
- CARROLL, Jeremy J., Christian BIZER, Pat HAYES et Patrick STICKLER (2005). “Named Graphs, Provenance and Trust”. *Proceedings of the 14th International Conference on World Wide Web*. Sous la dir. d’Allan ELLIS et Tatsuya HAGINO. WWW ’05. Chiba, Japan : Association for Computing Machinery, p. 613-622.
- COX, Simon, Chris LITTLE, Jerry R HOBBS et Feng PAN (mar. 2020). *Time Ontology in OWL*. <https://www.w3.org/TR/owl-time/>. W3C candidate recommendation. Dernière consultation : novembre 2020.
- DIALLO, Papa Fary, Olivier CORBY, Isabelle MIRBEL, Moussa LO et Seydina M. NDIAYE (juil. 2015). “HuTO : une Ontologie Temporelle Narrative pour les Applications du Web Sémantique”. *Ingénierie des Connaissances 2015*. Rennes, France.
- GHORBEL, Fatma, Fayçal HAMDY, Elisabeth MÉTAIS, Nebrasse ELLOUZE et Faiez GARGOURI (2018). “A Fuzzy-Based Approach for Representing and Reasoning

- on Imprecise Time Intervals in Fuzzy-OWL 2 Ontology”. *Natural Language Processing and Information Systems*. Sous la dir. de Max SILBERZTEIN, Faten ATIGUI, Elena KORNYSHOVA, Elisabeth MÉTAIS et Farid MEZIANE. Cham : Springer International Publishing, p. 167-178.
- GILROY, Amanda et Wilhelmus Maria VERHOEVEN (2000). *Epistolary histories : letters, fiction, culture*. USA : University of Virginia Press.
- GUTIERREZ, Claudio, Carlos A HURTADO et Alejandro VAISMAN (2006). “Introducing time into RDF”. *IEEE Transactions on Knowledge and Data Engineering* 19.2, p. 207-218.
- GUTIERREZ, Claudio, Carlos HURTADO et Alejandro VAISMAN (2005). “Temporal RDF”. *European Semantic Web Conference*. Sous la dir. d’Asunción GÓMEZ-PÉREZ et Jérôme EUZENAT. T. 3532. Springer, p. 93-107.
- HARTIG, Olaf (2017). “Foundations of RDF* and SPARQL* : (An alternative approach to statement-level metadata in RDF)”. *AMW 2017 11th Alberto Mendelzon International Workshop on Foundations of Data Management and the Web, Montevideo, Uruguay, June 7-9, 2017*. Sous la dir. de Juan REUTTER et Divesh SRIVASTAVA. T. 1912.
- KOSELLECK, Reinhartl (2002). *The Practice of Conceptual History : Timing History, Spacing Concepts*. Cultural Memory in the Present. Stanford, California : Stanford University Press.
- KRIEGER, Hans-Ulrich, Bernd KIEFER et Thierry DECLERCK (2008). “A Framework for Temporal Representation and Reasoning in Business Intelligence Applications”. *AAAI Spring Symposium : AI Meets Business Rules and Process Management*. Sous la dir. de Knut HINKELMANN, p. 59-70.
- LARROUSSE, Nicolas et Joël MARCHAND (juil. 2019). “A Techno-Human Mesh for Humanities in France : Dealing with preservation complexity”. *DH 2019*. Utrecht, Netherlands. URL : <https://hal.archives-ouvertes.fr/hal-02153016>.
- LASSILA, Ora, Ralph R SWICK et al. (1998). *Resource description framework (RDF) model and syntax specification*. <http://www.w3.org/TR/wd-rdf-syntax>. Dernière consultation : août 2020.
- MANOLA, Frank, Eric MILLER, Brian MCBRIDE et al. (2004). *RDF Primer*. <https://www.w3.org/TR/rdf-primer>. Dernière consultation : novembre 2020.
- MILEA, Viorel, Flavius FRASINCAR et Uzay KAYMAK (2011). “tOWL : a temporal web ontology language”. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 42.1, p. 268-281.
- NABONNAND, Philippe, éd. (1998). *La correspondance entre Henri Poincaré et Gösta Mittag-Leffler*. Basel : Birkhäuser.
- NGUYEN, Vinh, Olivier BODENREIDER et Amit SHETH (2014). “Don’t like RDF Reification ? Making Statements about Statements Using Singleton Property”. *Proceedings of the 23rd International Conference on World Wide Web*. Sous la dir. de Chin-Wan CHUNG. WWW ’14. Seoul, Korea : Association for Computing Machinery, p. 759-770.
- NGUYEN, Vinh, Olivier BODENREIDER, Krishnaprasad THIRUNARAYAN, Gang FU, Evan BOLTON, Núria QUERALT-ROSINACH, Laura Inés FURLONG, Michel

- DUMONTIER et Amit P. SHETH (2015). “On Reasoning with RDF Statements about Statements using Singleton Property Triples”. *Computer Research Repository* abs/1509.04513. URL : <http://arxiv.org/abs/1509.04513>.
- NOY, Natasha, Alan RECTOR, Pat HAYES et Chris WELTY (2006). *Defining n-ary Relations on the Semantic Web*. <https://www.w3.org/TR/swbp-n-aryRelations/>. W3C working group note. Dernière consultation : novembre 2020.
- POINCARÉ, Henri (1902). *La science et l'hypothèse*. Paris : Flammarion.
- (1905). *La Valeur de la Science*. Paris : Flammarion.
- (1908). *Science et Méthode*. Paris : Flammarion.
- PROST, Antoine (1996). *Douze Leçons sur l'histoire*. Points Histoire. Éditions du Seuil, Paris.
- PRUD'HOMMEAUX, Eric (2008). *SPARQL Query Language for RDF, W3C Recommendation*. Sous la dir. d'Eric PRUD'HOMMEAUX et Andy SEABORNE. <http://www.w3.org/TR/rdf-sparql-query/>. Dernière consultation : novembre 2020.
- ROLLET, Laurent (1997). “Autour de l'affaire Dreyfus. Henri Poincaré et l'action politique”. *Revue historique* 298, p. 49-101.
- éd. (2017). *La correspondance de jeunesse d'Henri Poincaré : les années de formation. De l'École polytechnique à l'École des Mines (1873-1878)*. Publications of the Henri Poincaré Archives. Basel : Springer International Publishing.
- SCHIAVON, Martina et Laurent ROLLET (2017). *Pour une histoire du Bureau des Longitudes (1795-1932)*. Nancy, France : Presses universitaires de Nancy-Éditions universitaires de Lorraine.
- SCHUELER, Bernhard, Sergej SIZOV, Steffen STAAB et Duc Thanh TRAN (2008). “Querying for Meta Knowledge”. *Proceedings of the 17th International Conference on World Wide Web*. Sous la dir. de Jinpeng HUAI, Robin CHEN, Hsiaowuen HON et Yunhao LIU. WWW '08. Beijing, China : Association for Computing Machinery, p. 625-634.
- TATON, René (1961). “Le XIX^e siècle”. *La science contemporaine*. Paris : Quadrige / Presses Universitaires de France.
- UDREA, Octavian, Diego Reforgiato RECUPERO et V. S. SUBRAHMANIAN (jan. 2010). “Annotated RDF”. *ACM Transactions on Computational Logic (TOCL)* 11.2, p. 1-41.
- WELTY, Chris et Richard FIKES (2006). “A Reusable Ontology for Fluents in OWL”. *Formal Ontology in Information Systems : Proceedings of the Fourth International Conference (FOIS 2006)*. Sous la dir. de BENNETT, BRANDON AND FELLBAUM, CHRISTIANE. T. 150. IOS Press, p. 226.
- ZHANG, Fu, Ke WANG, Zhiyin LI et Jingwei CHENG (2019). “Temporal Data Representation and Querying Based on RDF”. *IEEE Access* 7. Sous la dir. de Derek ABBOTT, p. 85000-85023.