Christophe Debruyne, Éamonn Clinton, and Declan O'Sullivan. Client-side Processing of GeoSPARQL Functions with Triple Pattern Fragments. In Sören Auer, Tim Berners-Lee, Christian Bizer, Sarven Capadisli, Tom Heath, Krzysztof Janowicz, and Jens Lehmann, editors, Workshop on Linked Data on the Web co-located with 26th International World Wide Web Conference (WWW 2017), CEUR Workshop Proceedings. CEUR-WS.org, 2017

Client-side Processing of GeoSPARQL Functions with Triple Pattern Fragments

Christophe Debruyne
ADAPT Centre
Trinity College Dublin
College Green
Dublin 2, Ireland
first.last@adaptcentre.ie

Éamonn Clinton
Ordnance Survey Ireland
Phoenix Park
Dublin 8, Ireland
first.last@osi.ie

Declan O'Sullivan
ADAPT Centre
Trinity College Dublin
College Green
Dublin 2, Ireland
first.last@adaptcentre.ie

ABSTRACT

"Place" is an important concept providing a useful dimension to explore, align and analyze data on the Linked Data Web. Though Linked Data datasets can use standardized geospatial predicates such as GeoSPARQL, access to SPARQL endpoints that supports these is not guaranteed. When not available, one needs to load the data into their own GeoSPARQL-enabled triplestores in order to avail of those predicates. Triple Pattern Fragments (TPF) is a proposal to make clients more intelligent in processing RDF, thereby lessening the burden carried by servers. In this paper, we propose to extend TPF to support GeoSPARQL. The contribution is a minimal extension of the TPF client that does not rely on a spatial database such that the extension can be run from within a browser. Even though our approach will unlikely outperform GeoSPARQL-enabled triplestores in terms of query execution time, we demonstrate its feasibility by means of a couple of use cases using data provided by `data.geohive.ie`, an initiative to publish authoritative, high-resolution geospatial data for The Republic of Ireland as Linked Data on the Web. This high-resolution data does cause a lot of network traffic, but related work showed how extending the communication between a TPF client and server reduces the number HTTP calls and some network traffic. The integration of our extension in one such optimization did reduce the overhead. We, however, decided to stick to our first implementation as it only extended the client in a minimal way. Future work includes investigating how our approach scales, and its usefulness of adding and using a spatial component to datasets.

CCS CONCEPTS

•Information systems → Resource Description Framework (RDF); Geographic information systems;

KEYWORDS

GeoSPARQL, Triple Pattern Fragments, Ordnance Survey Ireland

1 INTRODUCTION

Geospatial data is an important part of the Linked Data [3] Web, and this importance is demonstrated by the presence of numerous geographic datasets. Shadbolt et al. highlighted the importance of "place" in data and its role in interlinking and aligning datasets [14]. The importance of geospatial data is also reflected by the many (commercial) solutions that are available. Support for geospatial information in RDF is provided by commercial packages such as Stardog[1] and Oracle Spatial and Graph[2]. Academic prototypes include Parliament [2] and Strabon [10]. Geospatial information can thus act as a conduit for exploring and discovering information.

GeoNames[3] and LinkedGeoData[4] are examples of datasets that cover a vast part of the world. The Ordnance Survey Linked Data[5] and `data.geohive.ie` [5], on the other hand, provide geospatial information for Great Britain and The Republic of Ireland respectively. Some of these geographic datasets are authoritative, which means they can be trusted as being issued by an authority (such as a public administration). This is the case for the data provided by both Ordnance Surveys.

Datasets can rely on standardized vocabularies for representing and querying geospatial information. These vocabularies allow one to formulate queries with predicates that represent geospatial relations such as overlapping, part-of, disjoint, etc. The OGC (Open Geospatial Consortium) GeoSPARQL [13] standard, for example, not only defines a vocabulary for representing geospatial data on the Semantic Web, but also defines an extension to the SPARQL query language for processing that geospatial data.

The execution of geospatial queries may be computationally expensive; creating a load on the server and even disrupt it. In fact, people often provide data dumps and resolvable URIs as a "good enough" practice on the Linked Data Web in general to avoid this problem [17]. It is, however, unfortunate that one cannot avail of these geospatial predicates without loading the dumps into their own triplestores. The value of geospatial data, especially when they are authoritative, is when agents can engage with it; instead of analyzing the dump or crawling the data available on the frontend being able to formulate queries such as "give me all townlands in County Dublin."

[1] http://stardog.com/
[2] http://www.oracle.com/technetwork/database-options/spatialandgraph/overview/spatialandgraph-1707409.html
[3] http://www.geonames.org/
[4] http://linkedgeodata.org/
[5] http://data.ordnancesurvey.co.uk/

Some realized that query evaluation either happened on the server or on the client side and that there is a lack of options within that spectrum [17]. In [17], the authors proposed Triple Pattern Fragment (TPF), which provides a compromise by breaking down queries into simple queries (based on triple patterns) that the server needs to return and the client using these to compute the result set.

In this paper, we aim to investigate to what extent the notion of Triple Pattern Fragments can be extended to allow agents to engage with geospatial data. The contributions of this paper are i) an extension of the TPF client to support client-side processing of GeoSPARQL functions, and ii) a demonstration of the idea using authoritative geospatial information provided by the Ordnance Survey Ireland (OSi), Ireland's national mapping agency.

The remainder of this paper is organized as follows: Section 2 present `data.geohive.ie`, an initiative by the Ordnance Survey Ireland to publish authoritative high-resolution geospatial data as Linked Data on the Web the context in which the study in this paper has been conducted; Section 3 outlines our approach and implementation of client-side processing of GeoSPARQL queries by extending TPF; Section 4 is used to demonstrate our approach in the context of `data.geohive.ie`; Section 5 presents two initiatives within TCD where groups want to enrich their data with a geospatial component; in Section 6, we discuss some aspects of our study and look into integrating our approach with an optimized TPF server and client; and, finally, we conclude our paper in Section 7 and indicate the next steps.

2 DATA.GEOHIVE.IE

In [5], we reported on `data.geohive.ie`, which publishes and serves Ireland's authoritative boundary datasets – governed by the Ordnance Survey Ireland (OSi) – as Linked Data on the Web. This platform is the result of an ongoing collaboration between the OSi and ADAPT and currently serves information about administrative boundaries as these datasets were open to begin with. Fig. 1 depicts the geometry of County Dublin plotted on one of OSi's base maps.

The platform was designed to support two use cases; i) providing different "resolutions" of administrative boundaries and ii) providing the evolution of these boundaries as ordered by, for instance, Statutory Instruments. With "resolutions" we mean the level of detail in the geometries that represent the boundaries; the higher the resolution, the bigger the string representing the boundary and, as a consequence, the higher the overhead.

The first use case is supported by extending GeoSPARQL with concepts and relations specific to the OSi (e.g., "Townland" and "Electoral Division") and by using named graphs for each resolution. For the second use case, we extended PROV-O [12] with concepts such as "Statutory Instrument" (as a subclass of "Entity") and "Boundary Change" (as a subclass of "Activity").

One of the decisions made was to provide resolvable HTTP URIs and timely RDF dumps of the datasets, but no public access to the SPARQL endpoint. Availability is a concern for the OSi and we would rather host a limited instead of an unstable service. Though this is a situation we will reassess in the near future, we do recognize the potential of allowing agents – both human and computer-based – to explore the data with SPARQL. As a compromise, we provide a Triple Pattern Fragment (TPF [17]) server (and client). In short,

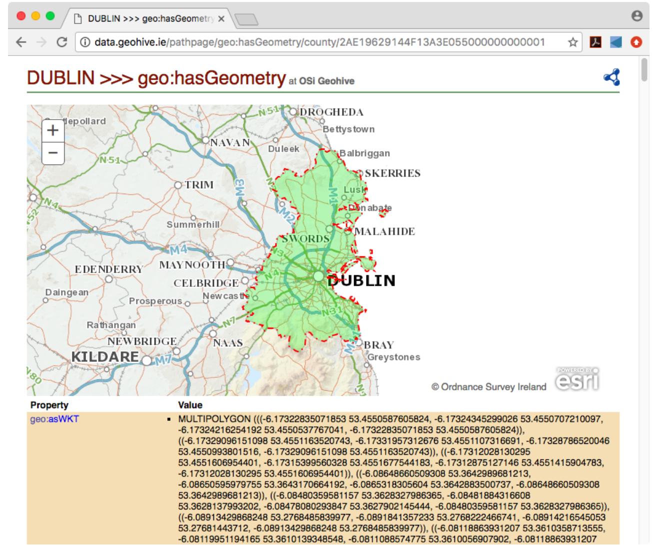

Figure 1: Plotting OSi's Polygon on OSi's base maps, part of the HTML served to users.

a TPF client breaks down a SPARQL query into multiple, simple queries and processes these as to compute the query result. This means that the client is tasked with joining, filtering, etc. the result set, decreasing the load required from the server. This, however, comes at the some costs including increased bandwidth caused by the communication between client and server, and slower query execution times.

A limitation, however, is that TPF does not provide support for geospatial predicates in SPARQL queries. This is because GeoSPARQL defines an extension of SPARQL that prescribes geospatial operators (such as "within", "overlaps", and "disjoint" – see Listing 1 for an example[6]) and these operators have not been implemented in TPF on either client or server side.

Listing 1: GeoSPARQL query for returning pairs labels in English of counties that are disjoint.

```
PREFIX osi: <http://ontologies.geohive.ie/osi#>
SELECT ?c1l ?c2l {
  ?c1 a osi:County .
  ?c1 rdfs:label ?c1l .
  ?c1 geo:hasGeometry ?g1 .
  ?c2 a osi:County .
  ?c2 rdfs:label ?c2l .
  ?c2 geo:hasGeometry ?g2 .

  FILTER (?c1 != ?c2)
  FILTER langMatches( lang(?c1l), "en" )
  FILTER langMatches( lang(?c2l), "en" )

  ?g1 geo:asWKT ?w1 .
  ?g2 geo:asWKT ?w2 .

  FILTER(geof:sfDisjoint(?w1, ?w2))
}
```

It is thus unfortunate that agents cannot avail of these predicates without them relying on ingesting the data in their own GeoSPARQL-enabled triplestores. While we know that geospatial

[6]Note that the namespaces for GeoSPARQL and its functions are omitted.

predicates can be computationally expensive, we will propose and investigate an extension of a TPF client that supports client-side processing of GeoSPARQL queries.

3 APPROACH AND IMPLEMENTATION

OSi's Linked Data relies on GeoSPARQL to represent features and geometries. OSi uses the Well-known Text (WKT) markup language for representing the geometries (such as polygons, multi-polygons, points representing centroids, etc.).[7] GeoSPARQL-enabled triplestores, or Geographic Information Systems in general, use these WKT representations of geometries to populate a database relying on data structures suitable for geospatial data such as R-Trees [11]. R-Trees, and similar data structures are used to index geometries such as points and polygons, which facilitates answering geospatial queries.

Our goal is to provide client-side processing of GeoSPARQL queries, and more precisely, GeoSPAQRL functions. One possible approach would be to store the geometries in a simple geospatial database on the client-side to compute the geospatial predicates. We, however, wanted to provide a solution that allowed third parties not only to avail of the geospatial predicates, but also did not require those parties to rely on additional components such as geospatial databases. The Node.JS implementation of the TPF client can, in fact, be run within a browser by bundling all the code and its dependencies into one JavaScript library. This would help us leverage engagement with OSi authoritative geospatial data and this created the additional requirement that the extension should solely rely on (code that can be bundled into) JavaScript. Of course we are aware of the limitations of computing GeoSPARQL queries in a browser on commodity hardware; browsers are not suitable to replace special-purpose triplestores. We will, however, discuss some of the issues later on in Section 5.

The different functions where implemented by interpreting the OGC standard and using set operators on the geometries. The following functions, to name a few, were implemented as follows:

geof:sfTouches. The intersection of the two geometries is not empty and only contains (a combination of) points or lines. If the intersection contains a polygon or a multi-polygon, the two geometries share an area.

geof:sfOverlaps. The intersection of the two geometries is not empty and should contain polygons or multi-polygons denoting areas.

geof:sfWithin. The intersection of the two geometries A and B is not empty, the difference between A and the intersection is empty, and the difference between B and the intersection contains geometries.

We note that the current implementation supports functions that we deem to occur often in examples. GeoSPARQL, in fact, prescribes a whole range of functions, some of which are more fine-grained. The function `geof:sfWithin`, for instance, covers both cases of a geometry being completely within (nTPP) another geometry and a geometry being within and touching the border

[7] We note that GeoSPARQL also prescribes another popular markup language for expressing geographical features and their geometry is the Geography Markup Language, or GML, which is an XML grammar defined by the Open Geospatial Consortium (OGC). For the time being, however, the OSi only serves WKT.

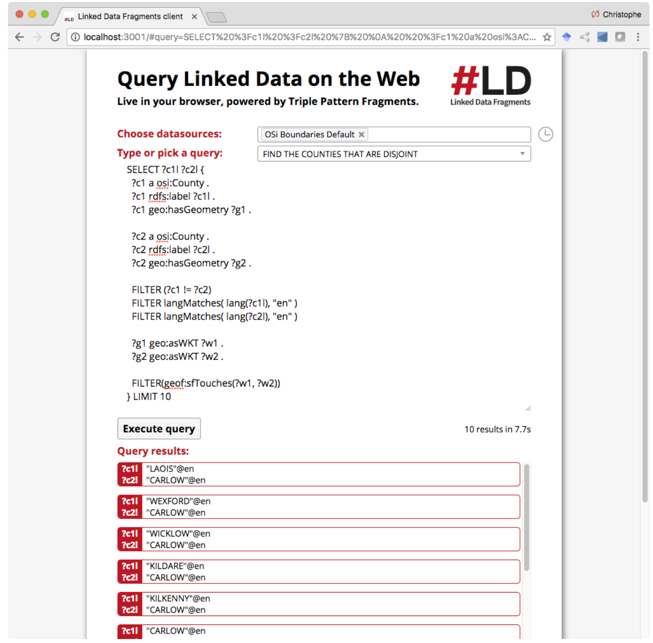

Figure 2: Retrieving the English labels of 10 pairs of Irish counties that touch each other – which means they share a part of their border

of another geometry (Tangential Proper Part – or TPP). Both are referred to with the predicates `geof:rcc8ntpp` and `geof:rcc8tpp`.

We extended V2.0.4 of the TPF Node.js Client [16]. Our extension is available on GitHub.[8] A web-client using this extension has also been made available online.[9] It relies on existing packages; one for converting WKT into GeoJSON [4], and one for manipulating GeoJSON objects.

4 DEMONSTRATION

For the purpose of our first demonstration, we use the Triple Pattern Fragment server set up for `data.geohive.ie`[10]. It contains description for various types of administrative boundaries such as counties, county (and/or) city councils, electoral divisions[11], etc.

Fig. 2 depicts a web client returning the English labels of 10 pairs of Irish counties who share a border, demonstrating that it supports the query listed in the previous section.

We have run the first query without the `LIMIT` clause 10 times on at 3 different points in time; morning, afternoon and evening – assuming there might be different loads on the network at different times. The client ran on a MacBook Pro 12.1 with an Intel Core i5 processor (2.7 GHz) and a memory of 8 GB (1867 MHz DDR3). The execution of the queries averaged at 126.140, 121.069, and 115.792 seconds – or about two minutes. Most of the processing time,

[8] https://github.com/chrdebru/Client.js
[9] http://theme-e.adaptcentre.ie/geo-tpf/
[10] http://vma01.adaptcentre.ie/
[11] The smallest legally defined administrative areas in the State for which Small Area Population Statistics (SAPS) are published from the Census.

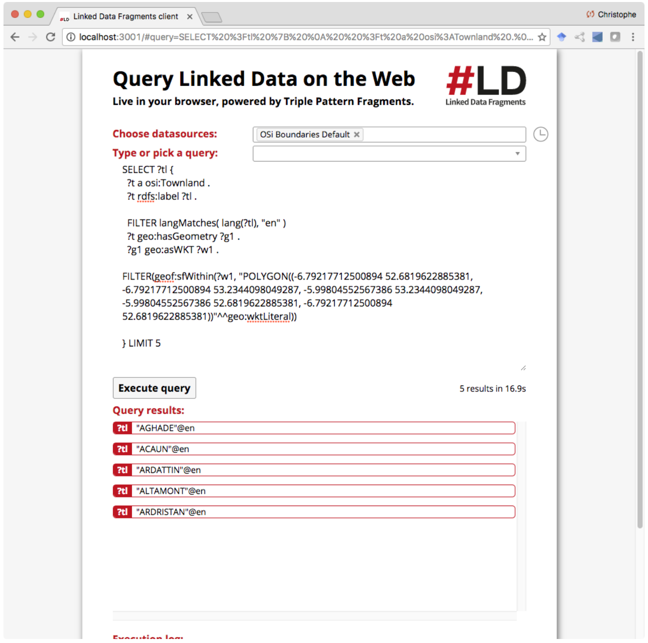

Figure 3: Retrieving the English labels of 5 townlands – one of the smaller administrative boundaries in Ireland – within a particular bounding box.

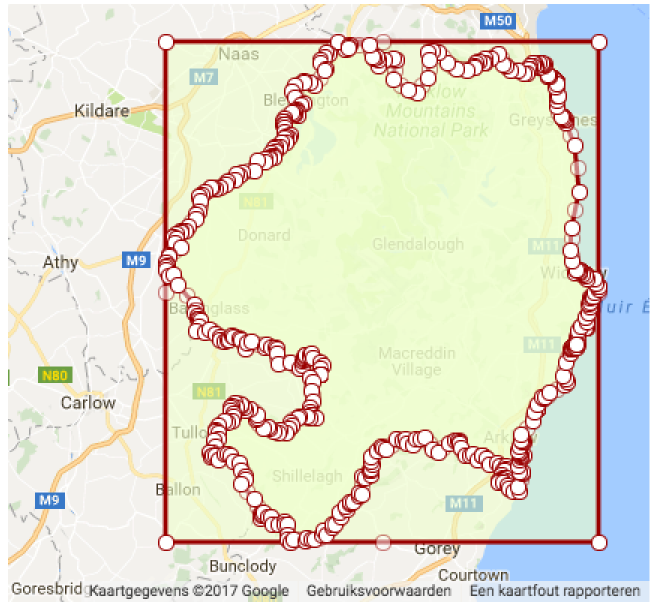

Figure 4: The bounding box of County Wicklow's border

however, went to computing the geospatial function instead of retrieving the data over the network.[12]

Fig. 3 shows the results of requesting five townlands that lie within County Wicklow's bounding box using the query from Listing 2 (in appendix). A (minimal) bounding box is a rectangle in which all points of that county's boundary reside. The boundary box is represented in WKT below and plotted – together with the boundary of its county – on a map in Fig. 4.

```
POLYGON((-6.79217712500894 52.6819622885381,
         -6.79217712500894 53.2344098049287,
         -5.99804552567386 53.2344098049287,
         -5.99804552567386 52.6819622885381,
         -6.79217712500894 52.6819622885381))
```

Note that the query using the boundary box will also return townlands that are outside of County Wicklow, yet within its bounding box.

5 USE CASES

So far, the demonstrators used the data made available by the Ordnance Survey Ireland. We will now proceed with two initiatives that enrich datasets with a geospatial component, subsequently used in combination with `data.geohive.ie` for exploring and querying the data with client-side processing of GeoSPARQL. The following use cases will provide examples of consulting different Triple Pattern Fragment servers to answer a query, i.e., examples of federated queries.

[12] The query returned 82 solutions in the result set, which corresponds to 41 pairs. We note, however, that there are some errors in the generalized dataset (see Section 6.5), but that the solutions are correct with respect to these errors.

5.1 TCD Library's Collections

Within Trinity College Dublin, the Library is investigating the adoption of Linked Data technologies to facilitate search, discovery and engagement with their collections and archives. Next to investigating appropriate methods and techniques for creating and managing Linked Data, they also investigate how their metadata can be enriched and contextualized geospatial data. Harry Clarke was an Irish stained-glass artist and book illustrator and many of his stained glasses can be admired in churches across Ireland. The Library's "Clarke Stained Glass Studios Collection" contains a wide variety of documents from stained glass designs and blueprints to correspondence. The library is currently digitizing these assets and aims to leverage user engagement with the collection.

The metadata – stored as Metadata Object Description Schema (MODS) – about this collection was transformed into RDF and links where created with an incomplete dataset of (mainly catholic) churches in Ireland of which the location is indicated with a point. A location-aware mobile application is currently being developed (see Fig. 5) that uses these points to direct users to churches where they can admire the stained glasses while reading the descriptions created by the archivists.

Fig. 6 demonstrates how we are able to retrieve assets related to churches that are located in County Dublin, using the query from Listing 3.

5.2 Sensor Data

Recently, the Chronic Disease Informatics Group (CDIG) in Trinity College Dublin is exploring ways to adopt semantic technologies to facilitate the combination and analysis of various heterogeneous data to, for instance, identify external factors that contribute to flare-ups of particular diseases. Data about patients are recorded at a particular place and time, and the locations of sensors – such

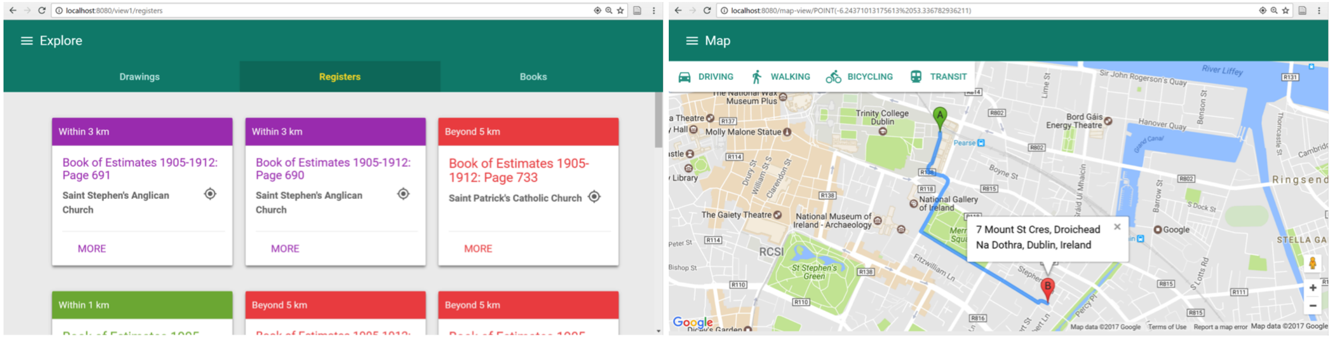

Figure 5: The location-aware application currently developed for the Harry Clarke Stained Glass Collection. The image on the left provides a glimpse of the assets for which a link with a church is available. Color-coding is used to show the distance between the user and the church. Links are provided that lead to a description of that asset, and to directions to the church (shown on the right).

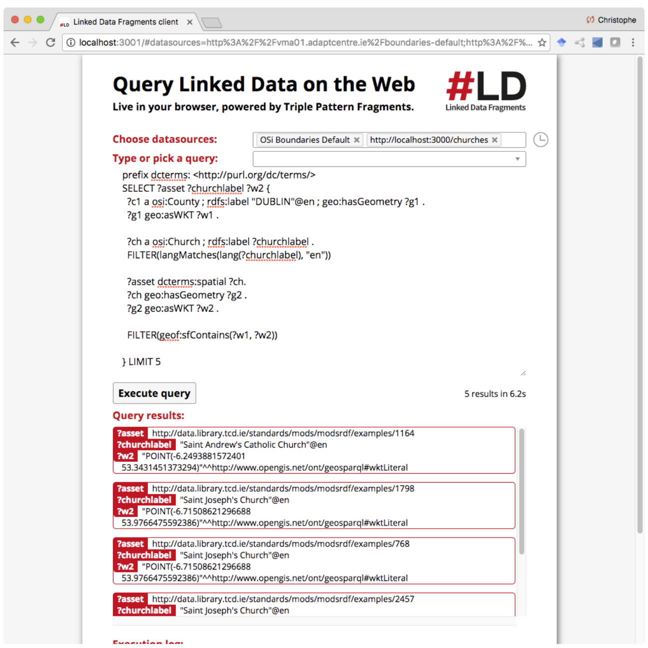

Figure 6: Obtaining assets from TCD Library that are related to churches in County Dublin.

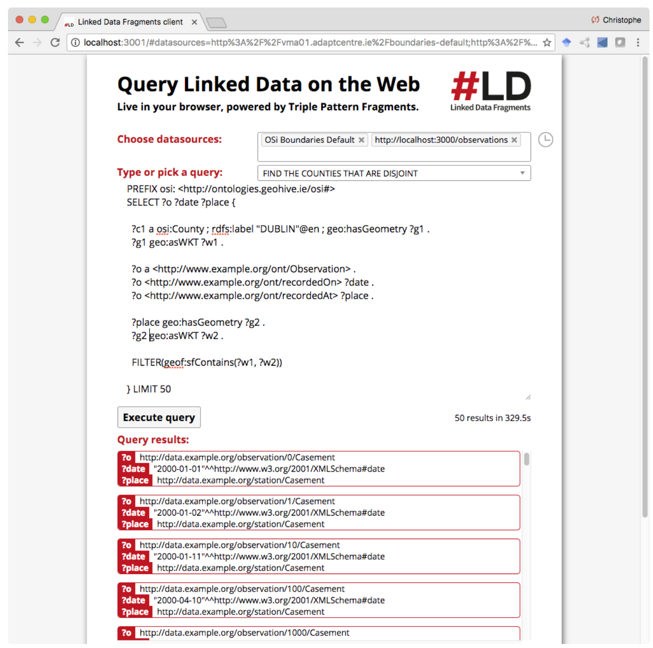

Figure 7: Retrieving observations in County Dublin.

as weather stations or air pollution detectors – are also known beforehand. One of the aspects the group wants to investigate is the notion of "space" that is present in their datasets.

For this demonstrator, we transformed a part of their data into RDF using R2RML, and generated WKT literals for the points in their datasets. We then proceeded to show how to formulate queries with GeoSPARQL, effectively showing that is straightforward to add and avail of a geospatial dimension to their data on one's machine, without the need to rely on bespoke triplestores.

Fig. 7 shows how one can retrieve observations in a particular County. Fig. 8 demonstrates how one can retrieve in which Electoral Divisions the weather stations are in, which may make sense if researchers want to relate observations with the smallest administrative unit used for the census. Listings 4 and 5 provides the queries used for aforementioned figures.

6 DISCUSSION

6.1 Client-side vs. Server-side Support for GeoSPARQL in TPF

We established that we aimed to extend a TPF client in Section 2 and its implementation details were described in Section 3. In this section, we will elaborate on this decision as well discuss the possible implications of extending the TPF server.

First, a TPF server is a server that complies with the specification laid out in [15]. No specifications for TPF clients exist; it is up one to

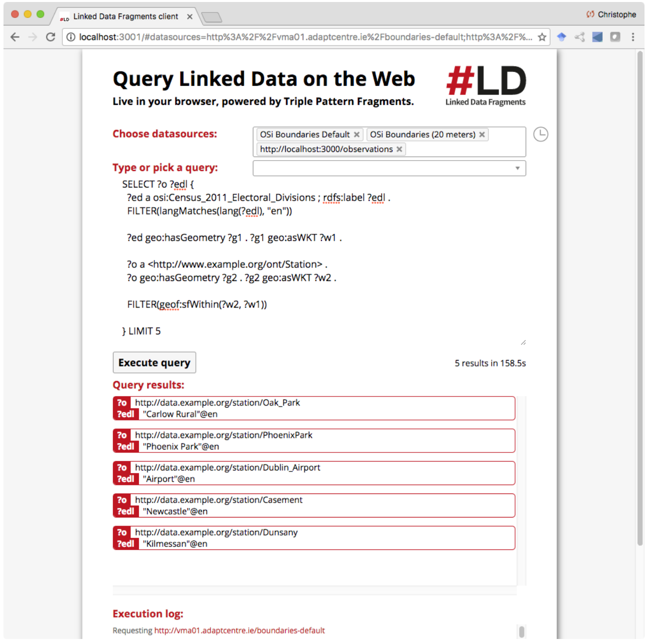

Figure 8: Obtaining the Electoral Divisions (EDs) where the weather stations are based using a GeoSPARQL function.

decide how a TPF client can engage with TPF servers. In that sense, we can argue that extending the client is less disruptive for the TPF initiative, and therefore an advantage over an implementation on the server-side.

TPF servers are defined as not to support SPARQL filter functions, though a study has shown it to be feasible to extend a TPF server with support for substring matching in filters with minimal impact on a server's load, though increase query response time [9]. We could envisage a similar approach for GeoSPARQL functions. Spatial predicates are, however, computationally expensive. Especially when taking into account complex geometries such as those published on `data.geohive.ie`. This could have an impact on a server's load when numerous clients interact with the server. This however, should be investigated as future work.

6.2 On the Limitations of a Client

The whole aim of this study was to propose a solution that would enable clients to easily process GeoSPARQL where GeoSPARQL-enabled endpoints would not be available. The TPF client/server architecture provided us with an ideal base the enable clients in doing so, rendering clients more intelligent in processing such data. We are, however, aware that computing these queries in JavaScript is not as efficient as relying on bespoke data structures and storage, especially if it was our goal to have such queries run in a browser environment. In Section 6.4, we will elaborate on related work on optimizing TPF on client and/or server side. But in future work, we should investigate how this approach scales with respect to query execution time and processing of large volumes of data, mainly due to the large geometries. Benchmarks such as Geographica [6] provide a starting point, but assuming that out approach will never scale as well as geospatial triplestores (which, we note, was never our intention to begin with), we would need to figure out what would constitute a "sensible" query; as in sufficiently specific to provide results in a reasonable time. This, however, will require investigating optimization techniques that we will now discuss.

6.3 On Performance

Using a Virtual Machine with 1GB of RAM and an Intel processor of 2.2 GHz on which is running Debian GNU/Linux 8.7 (jessie), we installed Parliament [2] (version 2.7.9 as the latest release with incompatible with the virtual machine) and loaded all 54,460 triples pertaining to the 100m generalization of the boundaries dataset on the disk, not in memory. The first query was run 10 times in a similar fashion and took, on average, 118.693 seconds to execute. With respect to the execution times on the client side, we deem our approach feasible. However, we already stated that our approach using JavaScript (in a browser) will unlikely outperform GeoSPARQL-enabled triplestores in terms of performance.

6.4 On Optimization

Though TPF allows for clients to become more intelligent by processing simple result sets based on triple patterns that requires minimal load for the server, bandwidth might become an issue; [7] noted that reducing server load comes at the price of a higher client-side load and increase in network load both in terms of HTTP requests and traffic. Several researchers proposed solutions to optimize the execution of queries with TPFs [1] [7] [8]. The authors of [1] and [8] investigated different TPF clients, and [7] investigated changes on both the TPF client and server.

OSi's geometries are large. Using the county boundary dataset (26 counties in total), the triples according to the pattern `{ ?geom geo:asWKT ?wkt }` result in RDF documents that contain 2.1MB, 2.8MB and 4.8MB of data for generalizations up to 100, 50 and 20 meters respectively. One can see that the TPF setup can generate a lot of traffic if the result set for `{ ?c1 a geo:Feature }` is joined with `{ ?c1 geo:hasGeometry ?g1 }`, where the latter corresponds with more than 54,000 triples for each resolution. The work presented in [7], proposing Bindings-Restricted Triple Pattern Fragments (brTPF), assumes that each triple pattern in a query would be used for joining, and bindings – values that will be used for joining – are communicated to an extended Triple Pattern Fragment Server that will reduce the result set of the next triple pattern based on those bindings. Though we have not conducted an extensive experiment comparing the two approaches, we did notice a decrease in HTTP requests when using brTPF.[13].

6.5 On the Boundary Datasets

While working on the boundary datasets, we noticed some topological inconsistencies currently hosted on `data.geohive.ie`. Where there should be 58 pairs of counties that border, for instance, we only have 41. The missing pairs shared some very small polygons next to lines and multi-lines. The borders of counties Carlow and Wicklow, for instance, share one tiny triangle of 0.000068034 square nanometers. The OSi has been made aware of errors that have crept

[13]We used the implementation referred to in [7], available at http://olafhartig.de/brTPF-ODBASE2016/

in the generalization of boundary data, and will rectify and release a new version of the datasets. Now this does no impact the contribution in this paper, but could confuse a reader comparing the result set with what can be observed on a map.

7 CONCLUSIONS AND FUTURE WORK

In this paper we presented a minimal extension of the Triple Pattern Fragments [17] client to support client-side processing of GeoSPARQL functions. This allows agents to avail of those functions when GeoSPARQL-enabled SPARQL endpoints are not available, or even when these endpoints do not provide support for these functions. One additional requirement was that the client should not rely on spatial databases as to provide a module that can run within a browser, further leveraging the use of spatial predicates. What we learned from this study is that it is feasible to delegate the responsibility of computing geospatial function to a TPF client.

The demonstrators presented in this paper all focused on the use of high-resolution data provided by `data.geohive.ie`, an initiative of the Ordnance Survey Ireland to publish their data as Linked Data. We furthermore elaborated on two initiatives, lead by different groups, in Trinity College Dublin aiming to add a geospatial component to their data. These two initiatives provide evidence that one can easily expose and combine their data with other datasets using GeoSPARQL.

Because the geometries are of high-resolution, the literals that capture these are large and network overhead is considerable. Integrating our approach with related work on optimizing the TPF client-server communication presented in [7] showed that some of the overhead can be reduced by extending both TPF server and client so that clients inform the server which terms will be used for joins. Another approach would have been to implement the filters on the server-side, a demonstrated by [9] for substring matching in filter clauses. We, however, currently favor our first implementation, as it is only an extension of the client. Though there is evidence that the former reduces to some extent overhead, support for GeoSPARQL functions in filters on the server side should be investigated in the future.

Finally, we furthermore aim to complete the set of functions prescribed by GeoSPARQL and conduct studies to study the usability and usefulness of our approach, in the broadest sense of the word, involving different types of stakeholders.

Acknowledgements

We thank the Ordnance Survey Ireland (OSi) for permitting us to use their boundaries dataset for the purposes of this research project. Within OSi, we are especially grateful for the input and domain expertise provided by Lorraine McNerney. The ADAPT Centre for Digital Content Technology is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund. We furthermore thank Brian Reddy, Mark Little, and Brett Houlding from the Chronic Disease Informatics Group (CDIG) in Trinity College Dublin for allowing us to use their data as part of a demonstrator in this paper. We thank The Library of Trinity College Dublin and Peru Bhardwaj for access to their data and mobile application. Finally, we would like to thank the (anonymous) reviewers – and in particular Ruben Verborgh – for their valuable comments, and Olaf Hartig for clarifying some aspects of brTPF.

REFERENCES

[1] Maribel Acosta and Maria-Esther Vidal. 2015. Networks of Linked Data Eddies: An Adaptive Web Query Processing Engine for RDF Data. In *The Semantic Web - ISWC 2015 - 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part I (Lecture Notes in Computer Science)*, Marcelo Arenas, Óscar Corcho, Elena Simperl, Markus Strohmaier, Mathieu d'Aquin, Kavitha Srinivas, Paul T. Groth, Michel Dumontier, Jeff Heflin, Krishnaprasad Thirunarayan, and Steffen Staab (Eds.), Vol. 9366. Springer, 111–127. DOI: http://dx.doi.org/10.1007/978-3-319-25007-6_7

[2] Robert Battle and Dave Kolas. 2012. Enabling the geospatial Semantic Web with Parliament and GeoSPARQL. *Semantic Web* 3, 4 (2012), 355–370. DOI: http://dx.doi.org/10.3233/SW-2012-0065

[3] Christian Bizer, Tom Heath, and Tim Berners-Lee. 2009. Linked Data - The Story So Far. *Int. J. Semantic Web Inf. Syst.* 5, 3 (2009), 1–22. DOI: http://dx.doi.org/10.4018/jswis.2009081901

[4] Howard Butler, Martin Daly, Allan Doyle, Sean Gillies, Stefan Hagen, and Tim Schaub. 2016. *Request for Comments: 7946 – The GeoJSON Format.* Request for Comments. Internet Engineering Task Force (IETF). https://tools.ietf.org/html/rfc7946.

[5] Christophe Debruyne, Éamonn Clinton, Lorraine McNerney, Atul Nautiyal, and Declan O'Sullivan. 2016. Serving Ireland's Geospatial Information as Linked Data. In *Proceedings of the ISWC 2016 Posters & Demonstrations Track co-located with 15th International Semantic Web Conference (ISWC 2016), Kobe, Japan, October 19, 2016. (CEUR Workshop Proceedings)*, Takahiro Kawamura and Heiko Paulheim (Eds.), Vol. 1690. CEUR-WS.org. http://ceur-ws.org/Vol-1690/paper14.pdf

[6] George Garbis, Kostis Kyzirakos, and Manolis Koubarakis. 2013. Geographica: A Benchmark for Geospatial RDF Stores (Long Version). In *The Semantic Web - ISWC 2013 - 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part II (Lecture Notes in Computer Science)*, Harith Alani, Lalana Kagal, Achille Fokoue, Paul T. Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha F. Noy, Chris Welty, and Krzysztof Janowicz (Eds.), Vol. 8219. Springer, 343–359. DOI: http://dx.doi.org/10.1007/978-3-642-41338-4_22

[7] Olaf Hartig and Carlos Buil Aranda. 2016. Bindings-Restricted Triple Pattern Fragments. In *On the Move to Meaningful Internet Systems: OTM 2016 Conferences - Confederated International Conferences: CoopIS, C&TC, and ODBASE 2016, Rhodes, Greece, October 24-28, 2016, Proceedings (Lecture Notes in Computer Science)*, Christophe Debruyne, Hervé Panetto, Robert Meersman, Tharam S. Dillon, eva Kühn, Declan O'Sullivan, and Claudio Agostino Ardagna (Eds.), Vol. 10033. Springer, 762–779. DOI: http://dx.doi.org/10.1007/978-3-319-48472-3_48

[8] Joachim Van Herwegen, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. 2015. Query Execution Optimization for Clients of Triple Pattern Fragments. In *The Semantic Web. Latest Advances and New Domains - 12th European Semantic Web Conference, ESWC 2015, Portoroz, Slovenia, May 31 - June 4, 2015. Proceedings (Lecture Notes in Computer Science)*, Fabien Gandon, Marta Sabou, Harald Sack, Claudia d'Amato, Philippe Cudré-Mauroux, and Antoine Zimmermann (Eds.), Vol. 9088. Springer, 302–318. DOI: http://dx.doi.org/10.1007/978-3-319-18818-8_19

[9] Joachim Van Herwegen, Laurens De Vocht, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. 2015. Substring Filtering for Low-Cost Linked Data Interfaces. In *The Semantic Web - ISWC 2015 - 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part I (Lecture Notes in Computer Science)*, Marcelo Arenas, Óscar Corcho, Elena Simperl, Markus Strohmaier, Mathieu d'Aquin, Kavitha Srinivas, Paul T. Groth, Michel Dumontier, Jeff Heflin, Krishnaprasad Thirunarayan, and Steffen Staab (Eds.), Vol. 9366. Springer, 128–143. DOI: http://dx.doi.org/10.1007/978-3-319-25007-6_8

[10] Kostis Kyzirakos, Manos Karpathiotakis, and Manolis Koubarakis. 2012. Strabon: A Semantic Geospatial DBMS. In *The Semantic Web - ISWC 2012 - 11th International Semantic Web Conference, Boston, MA, USA, November 11-15, 2012, Proceedings, Part I (Lecture Notes in Computer Science)*, Philippe Cudré-Mauroux, Jeff Heflin, Evren Sirin, Tania Tudorache, Jérôme Euzenat, Manfred Hauswirth, Josiane Xavier Parreira, Jim Hendler, Guus Schreiber, Abraham Bernstein, and Eva Blomqvist (Eds.), Vol. 7649. Springer, 295–311. DOI: http://dx.doi.org/10.1007/978-3-642-35176-1_19

[11] Yannis Manolopoulos, Alexandros Nanopoulos, Apostolos N. Papadopoulos, and Yannis Theodoridis. 2006. *R-Trees: Theory and Applications.* Springer. DOI: http://dx.doi.org/10.1007/978-1-84628-293-5

[12] Deborah McGuinness, Timothy Lebo, and Satya Sahoo. 2013. *PROV-O: The PROV Ontology.* W3C Recommendation. W3C. http://www.w3.org/TR/2013/REC-prov-o-20130430/.

[13] Matthew Perry and John Herring. 2012. *GeoSPARQL - A Geographic Query Language for RDF Data.* OGC Standard. OGC. http://www.opengeospatial.org/standards/geosparql.

[14] Nigel Shadbolt, Kieron O'Hara, Tim Berners-Lee, Nicholas Gibbins, Hugh Glaser, Wendy Hall, and M. C. Schraefel. 2012. Linked Open Government Data: Lessons from Data.gov.uk. *IEEE Intelligent Systems* 27, 3 (2012), 16–24. DOI: http://dx.doi.org/10.1109/MIS.2012.23

[15] Ruben Verborgh. 2017. *Triple Pattern Fragments: A low-cost, queryable Linked Data Fragments interface.* Unofficial Draft. Hydra W3C Community Group. http://www.hydra-cg.com/spec/latest/triple-pattern-fragments/.

[16] Ruben Verborgh and Miel Vander Sande. 2016. LinkedDataFragments/Client.js: v2.0.4. (Dec 2016). DOI: http://dx.doi.org/10.5281/zenodo.216592

[17] Ruben Verborgh, Miel Vander Sande, Olaf Hartig, Joachim Van Herwegen, Laurens De Vocht, Ben De Meester, Gerald Haesendonck, and Pieter Colpaert. 2016. Triple Pattern Fragments: A low-cost knowledge graph interface for the Web. *J. Web Sem.* 37-38 (2016), 184–206. DOI: http://dx.doi.org/10.1016/j.websem.2016.03.003

A QUERIES

Here we list the queries we have used for our demonstration. Queries listed in Listings 1 and 2 can be run against the TPF server set up for `data.geohive.ie`. The remaining queries, however, depend on data we cannot make available.

Listing 2: Townlands query of Fig. 3.

```
PREFIX osi: <http://ontologies.geohive.ie/osi#>
SELECT ?tl {
  ?t a osi:Townland .
  ?t rdfs:label ?tl .

  FILTER langMatches( lang(?tl), "en" )
  ?t geo:hasGeometry ?g1 .
  ?g1 geo:asWKT ?w1 .

  FILTER(geof:sfWithin(?w1, "POLYGON((-6.79217712500894
        52.6819622885381, -6.79217712500894 53.2344098049287,
        -5.99804552567386 53.2344098049287, -5.99804552567386
        52.6819622885381, -6.79217712500894 52.6819622885381))"^^
        geo:wktLiteral))
} LIMIT 5
```

Listing 3: Library assets query of Fig. 6

```
PREFIX osi: <http://ontologies.geohive.ie/osi#>
PREFIX dcterms: <http://purl.org/dc/terms/>
SELECT ?asset ?churchlabel ?w2 {
  ?c1 a osi:County .
  ?c1 rdfs:label "DUBLIN"@en .
  ?c1 geo:hasGeometry ?g1 .
  ?g1 geo:asWKT ?w1 .

  ?ch a osi:Church .
  ?ch rdfs:label ?churchlabel .
  FILTER(langMatches(lang(?churchlabel), "en"))

  ?asset dcterms:spatial ?ch.
  ?ch geo:hasGeometry ?g2 .
  ?g2 geo:asWKT ?w2 .

  FILTER(geof:sfContains(?w1, ?w2))
} LIMIT 5
```

Listing 4: Observations in Dublin query of Fig. 7.

```
PREFIX osi: <http://ontologies.geohive.ie/osi#>
SELECT ?o ?date ?place {
  ?c1 a osi:County .
  ?c1 rdfs:label "DUBLIN"@en .
  ?c1 geo:hasGeometry ?g1 .
  ?g1 geo:asWKT ?w1 .

  ?o a <http://www.example.org/ont/Observation> .
  ?o <http://www.example.org/ont/recordedOn> ?date .
  ?o <http://www.example.org/ont/recordedAt> ?place .

  ?place geo:hasGeometry ?g2 .
  ?g2 geo:asWKT ?w2 .

  FILTER(geof:sfContains(?w1, ?w2))
} LIMIT 50
```

Listing 5: Electoral Divisions query of Fig. 8.

```
PREFIX osi: <http://ontologies.geohive.ie/osi#>
SELECT ?o ?edl {
  ?ed a osi:Census_2011_Electoral_Divisions .
  ?ed rdfs:label ?edl .
  FILTER(langMatches(lang(?edl), "en"))

  ?ed geo:hasGeometry ?g1 .
  ?g1 geo:asWKT ?w1 .

  ?o a <http://www.example.org/ont/Station> .
  ?o geo:hasGeometry ?g2 .
  ?g2 geo:asWKT ?w2 .

  FILTER(geof:sfContains(?w1, ?w2))
} LIMIT 5
```