Zaragozí B., Gutierrez A., Trilles S. (2021) Analysis of Public Transport Mobility Data: A System for Sharing and Reusing GIS Database Queries. In: Grueau C., Laurini R., Ragia L. (eds) *Geographical Information Systems Theory, Applications and Management. GISTAM 2020.* Communications in Computer and Information Science, vol 1411. Springer, Cham. https://doi.org/10.1007/978-3-030-76374-9_7

Analysis of public transport mobility data: a system for sharing and reusing GIS database queries*

Benito Zaragozí[1 [0000−0003−2501−484X]], Aaron Gutierrez[1 [0000−0003−0557−6319]], and Sergio Trilles[1 [0000−0002−9304−0719]]

[1] Departament de Geografia, Universitat Rovira i Virgili, C/Joanot Martorell, Vilaseca, Spain
[2] Institute of New Imaging Technologies, Universitat Jaume I, Av. Vicente Sos Baynat s/n, Castellón de la Plana, Spain

Abstract. Data from automated fare collection systems have become almost essential in the study of the mobility of people using public transport. Among other advantages, the data collected enable longitudinal studies to be carried out with a detail that other sources cannot approximate. However, despite the great potential of these data, the data collecting systems are usually intended for purely accounting purposes and not for carrying out mobility studies. Largely for this reason, these data are not always used to their full potential, and so it is necessary to propose strategies that allow the preparation and exploitation of these data, especially in those cases where the usefulness and value of the data have not yet been proven. This study proposes a workflow that seeks to prevent duplication of efforts when querying this type of data. The implementation of a generic database model and a protocol for sharing meaningful queries and results greatly facilitates an initial analysis of these data. This strategy has been applied within a specific project, but it could be the basis for sharing methods between different studies.

Keywords: public transportation · automated fare collection system · smart card data · domain-specific language · file naming convention.

* Research funded by the Spanish Ministerio de Ciencia e Innovación [grant number CSO2017-82156-R], the AEI/FEDER,UE, the Departament d'Innovació, Universitats i Empresa, Generalitat de Catalunya [grant number 2017SGR22] and the Escola d'Administració Publica de Catalunya, Generalitat de Catalunya [grant number 2018 EAPC 00002]. Sergio Trilles has been funded by the Juan de la Cierva - postdoctoral programme of the Ministry of Science and Innovation - Spanish Government (IJC2018-035017-I)

1 Introduction

In recent decades, automated fare collection systems (AFCSs) around the world have been generating vast amounts of data. In many cities and metropolitan regions, the public use smart transport cards or similar technologies each time they board public transport (bus, train, or metro) and the AFCS collects data from its validation. The main aim is for accounting purposes, and so the data is not organised for analysis, nor are there specific tools for exploring the data to its full potential. Despite not being its main purpose, these data have been analysed to create relevant knowledge for understanding the behaviour of users and generating better quality services [15]. These types of analysis are widespread in the scientific literature [20, 2]. Some works use smart transport card validations to achieve huge flexibility for studying any temporal and geographical aspect [19]. An example is the use of smart travel card data to detect different profiles of public transport users [18], build origin-destination matrices [1], and investigate user mobility patterns [17].

Many authors agree on the opportunities that smart travel card data provide for transport and mobility studies [15, 20]. However, these data are challenging to handle and costly to analyse. Smart transport cards gather all transactions, and so the size of data may become huge after a relatively short period. The managers of public transport systems in large cities or regions may have the resources and the will to analyse such data to obtain some value. However, in regions or cities with fewer resources, analysing this data may not be a priority, especially if the value of analysing the data has yet to be demonstrated. In these cases, collaboration between public transport authorities and mobility research groups in universities has become an interesting strategy to start analysing data from AFCSs [25].

There are no standard solutions for analysing data from an AFCS. During the last decade, many research works have applied differing technologies or approaches to analyse different amounts of data generated by an AFCS. For example, the authors in [21] used *big data* technologies to analyse 160 million records from the Jakarta's Bus Rapid Transit in Indonesia, and another study analysed nearly 200 GB of data logs from the AFCS in the city of Montevideo in Uruguay[7]. Other studies were more focussed on real-time analysis and developed tailored solutions such as the data mining frameworks for bus service management [3]. Some studies did not need to analyse such volumes of data and used better-known software for working (e.g. MS Excel, SPSS, QGIS, Rstudio) [8, 9]. Finally, there are examples of other studies that did not indicate the use of any specific technology to perform the analysis [22]. Among this diversity of options, it can be highlighted that SQL databases are widely used to analyse this data, alone or in combination with other tools. In a recent systematic literature review, seven out of nine of the documents that reported the tools chosen for the analysis of smart travel card data used an SQL database [16].

1.1 A collaboration for analysing data from an AFCS

This research work is the result of a research project carried out jointly by a research group at the Universitat Rovira i Virgili (Tarragona, Spain) and the Territorial Mobility Authority of the Camp de Tarragona (ATMCdT, according to its acronym in Catalan). The ATMCdT has been running an AFCS for more than ten years, while serving an area of 2,998 km^2 and a population of 626,277 residents [14]. The area includes 132 municipalities and 457 interurban bus stops. Figure 1 shows how the population is unequally distributed over the study area. This region is shaped by coastal tourism, which further increases the pressure on public services in the largest cities and municipalities where tourism plays a prominent role. This drives an unbalanced public transport demand, especially during the summer season.

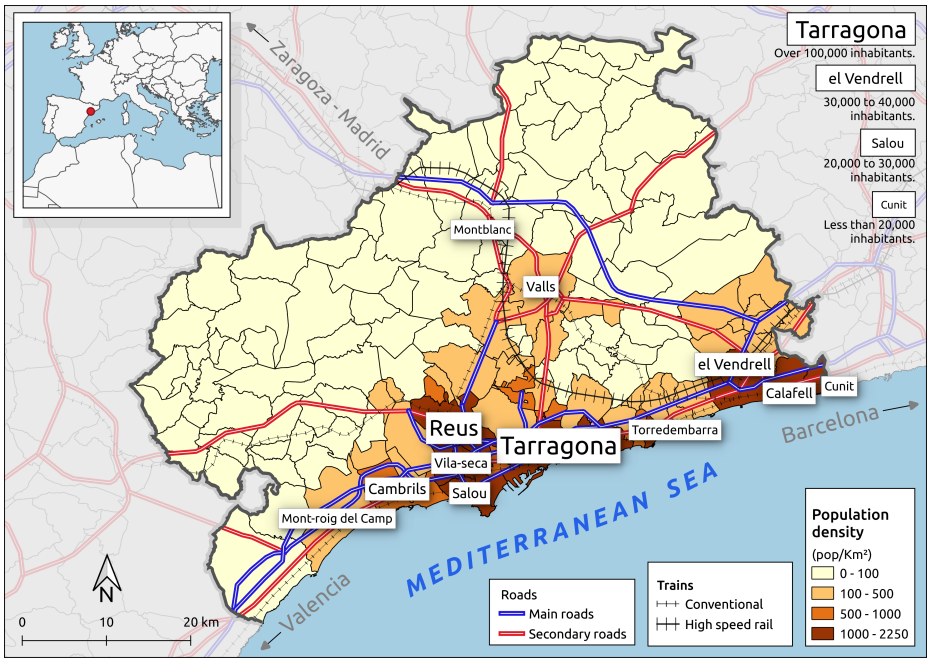

Fig. 1: Reference map and context of the the Territorial Mobility Authority of Camp de Tarragona (ATMCdT) service area.

During the last decade, ATMCdT has been using the data from its AFCS for creating reports for different needs (such as network planning, accounting, and management of public grants). These tasks are performed through tailored queries directly exploding the raw data logs. In this way, the fundamental purpose of the data collected by the AFCS is fulfilled. More recently, in 2017 the collaboration mentioned above started several studies for analysing the effective-

ness and spatial coverage of the public transport system [5], and the use of public transport by tourists [11, 12, 6]. A subsequent study has found evidence of the different patterns of public transport use by tourists in the summer [10]. These studies also demonstrated that smart card data collected by the ATMCdT has much to offer.

Together with the wide variety of technical options discussed above, the design of a well-defined workflow and the choice of flexible free and open software tools can make the initial exploitation of this data less expensive and avoid duplicating efforts [27]. However, although the studies mentioned in the previous paragraph faced data management problems, documenting these problems was far from their main goals, and so data preparation was not documented in detail.

In an initial effort to document the management and analysis of these data, a system was proposed to uniquely name the AFCS data queries. This system enables the creation of a repository to store the code together with the results, thereby facilitating collaboration within the same research team and opening the possibility of contributing to a public repository where these methods can be shared [26]. Figure 2 shows a diagram that describes the communication between the different roles of the team, the encapsulated access to a structured database, and the need for a results repository containing spreadsheets or geographic information system (GIS) files with specific and descriptive names. The main advantages of this approach are that nobody need repeat the same query and the database manager will hold a useful base of code for building new queries [26].

1.2 Objectives

As mentioned above, the duplication – or multiplication – of efforts and the strategy adopted to tackle this was first described in [26]. However, this work describes the proposed framework (database schema and naming convention) and provides more detailed examples in which the advantages of sharing methods and SQL queries become more evident.

The main aim of this work is to offer the lessons learned in this project when the data from an AFCS was analysed for the first time. To achieve this, this chapter develops the following objectives:

1. Study the data gathered by the ATMCdT and design a simple but adequate GIS database model. This model includes only the most basic mobility data that could be collected by any AFCS.
2. Propose a domain-specific language (DSL) to be used as a file naming convention that enhances code reuse and time saving. This should allow the description of the largest possible number of queries of this domain.
3. Extend a previous mobility grammar presented in [26] to add the capability to store GIS results.
4. Apply and evaluate the proposed strategy, showing detailed examples that demonstrate the value of code sharing in this area.

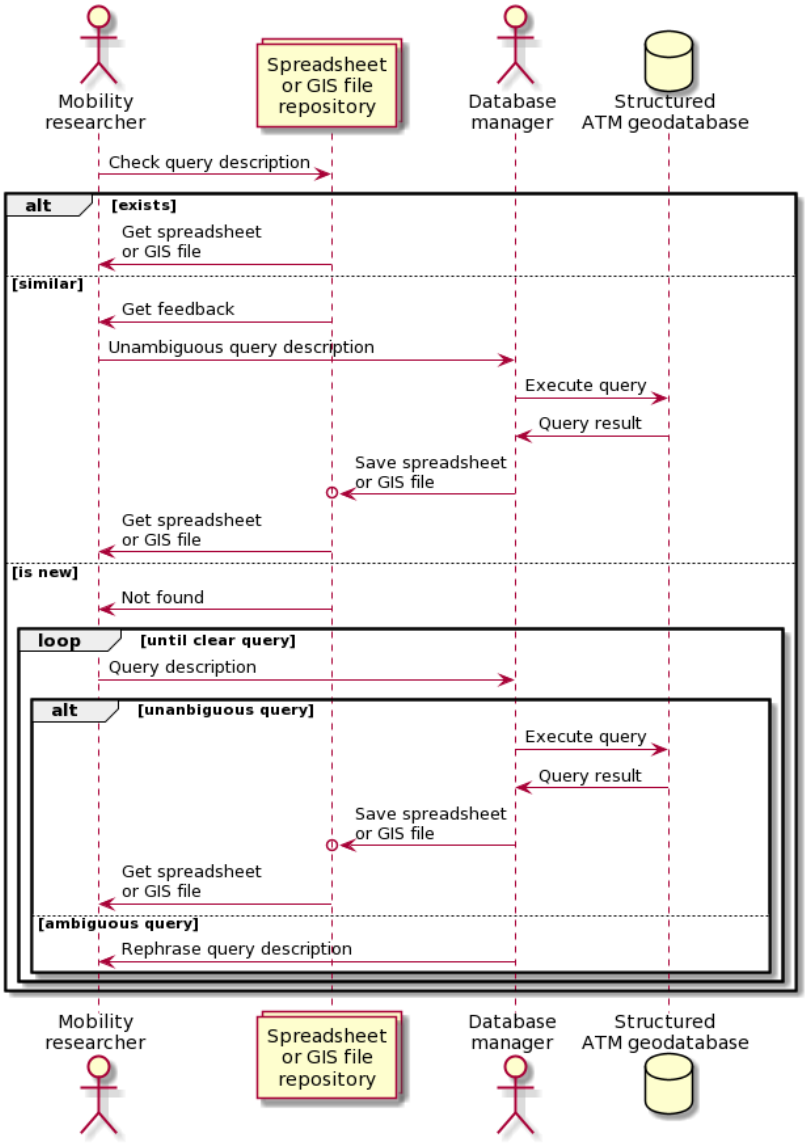

Fig. 2: Sequence diagram showing the proposed workflow. The spreadsheet or GIS file repository works on the basis of a file naming convention. Adapted from [26].

This chapter is organised as follows. The next section describes the proposed framework, which includes a logical database model and a naming convention for unambiguously storing the database queries. Section 3 shows how the framework can be used to store database queries and reuse them to avoid duplicating ef-

forts. Three compelling examples are described. Finally, Section 4 presents some concluding remarks and indicates issues that will be addressed in future works.

2 Proposed framework

2.1 GIS database schema

The analysis of the ATMCdT data logs reveals some heterogeneity in the structure and attributes that are meaningless for research purposes. Avoiding this complexity is essential so that the different analyses can be reproducible and extrapolated to other similar projects. This can be achieved by adding an abstraction layer for querying the data more quickly, and this can be done by using a common data model. Depending on the type of bus tickets, the data collected by the ATMCdT includes between 22 and 60 different attributes (columns). Many of these attributes are of no interest for analysing mobility patterns but, as said before, they fulfil an accounting purpose. Considering the needs of the project, only a few attributes are beneficial for analysing mobility patterns and user profiles in the region: the exact day and time of travel; the id of the stop where the passenger boarded; the company and carrier that operate the transport; the municipality; and the type of fare used in each transaction. The destination stop can sometimes also be registered. The main advantage of these data is that the information has a spatio-temporal dimension, and it enables us to perform cross-sectional studies as well as longitudinal analysis [10]. These databases do not store much data on the socio-economic profile of travellers, and these data cannot be used due to legal constraints.

The design of the database is quite generic and is given by the characteristics of the raw data collected by the ATMCdT (see Figure 3). This model does not include other transactions that are also systematically recorded (such as card sales, recharges, or cancellations). For clarity, the tables were named following the main elements of the general traffic feed specification (GTFS) and the columns were named predictably. The level of specificity of this model enables collecting the most basic information for analysing the mobility of users. This information is presumably available in all AFCSs, and it would take a relatively simple extract-transform-load (ETL) process to structure the data in this way. Figure 3 shows that, in addition to the ATMCdT data, the database was enriched with some layers of geographical information (such as municipalities, roads, shoreline, population, and land uses). The stop locations and routes were manually geolocated and digitised. However, in a geospatial database, there is room for increasing the list of support data layers by applying spatial join operations.

2.2 *MobilityFNC* definition

One of the most significant difficulties detected in the proposed workflow (Figure 2) is the query definition process between the mobility researcher and the

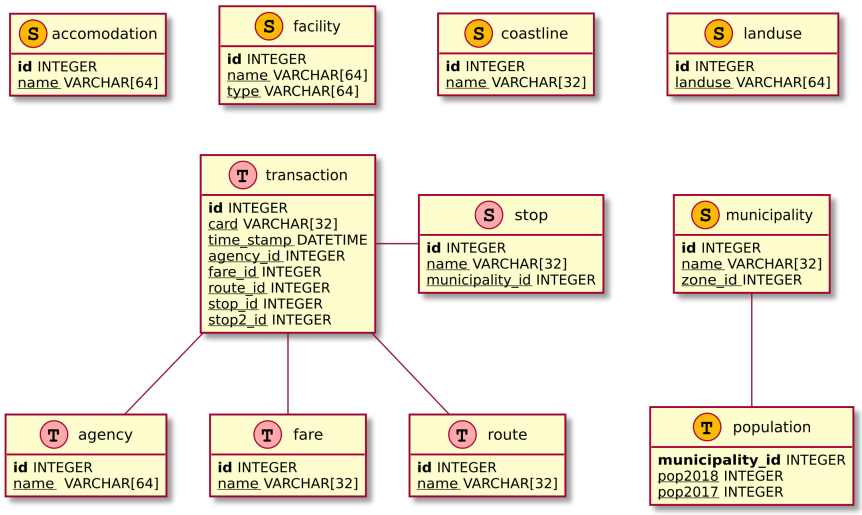

Fig. 3: Entity-relationship data model derived from the ATMCdT smart card log data. The diagram distinguishes between regular attribute tables (T), spatial tables (S) and the colours indicate if the data comes from ATMCdT (red) or an external source (orange). Based on [26].

database administrator. The lack of a common language can cause loss of information and context, which increases the probability of needing more iterations before obtaining the desired query. An approach to resolve this problem is to define a domain-specific language (DSL) for this type of application need. A DSL is considered a programming language and defines a set of notations and abstractions to cover a particular problem domain [23]. The main advantage of designing and using a DSL is that they offer more significant optimisation and adaptation to the particular domain and systematic reuse [4, 23]. DSLs have been used previously in different application domains (HTML, Unix shell scripts, and GraphViz are widely used examples). A DSL does not always fit a specific application domain. For example, SQL is recognised as a DSL for managing databases, but it remains a very general and comprehensive language [13].

In this work, the previous mobility file naming convention (*MobilityFNC*) introduced in [26] is expanded. *MobilityFNC* is used as a convention to represent and define queries accurately. The main objective is to improve the communication between the two most important roles: mobility researcher and database manager. This DSL specifies a structured and understandable method for mobility specialists to make a database query without any previous experience in database languages. *MobilityFNC* is used for encoding the name files of the SQL scripts. A mobility researcher who uses this DSL can immediately recognise if the query has been made previously or if there is a similar query with useful

code to create a new one. Although *MobilityFNC* can be extended to several kinds of public mobility, such as cars, trains, or planes, based on the ATMCdT scenario, *MobilityFNC* has been applied to public bus mobility. In the same way as SQL, *MobilityFNC* is used to represent the *shape* and main parts of a query result. Thus, the main intention is to maximise its compatibility and improve this proposal so that it can be translated into valid SQL, at least for a previously known database model (like the model shown in Figure 3).

The query definition and results are stored in two different files and both files are named using this DSL. In this way, the query definition and query result guarantee consistency – the SQL query defined by the database experts and the results file – are inseparable. This repository should be checked before developing a new query.

Figure 2 presents the workflow enhanced by using *MobilityFNC*. The largest difference compared to the version presented in [26] is the option to return and store the results as GIS files. *MobilityFNC* can act as a queries repository and joins SQL scripts and results following a filename nomenclature. Figure 2 shows these three situations: 1) the query exists when a query description was previously encoded, a mobility expert can obtain the results without any other procedure; 2) a similar query exists if the query to encode is similar to another other previously executed, the database manager can adapt it and obtain the results; and 3) in other cases (when similar queries do not exist) the database specialist follows the query description encoded using *MobilityFNC* and creates and executes the SQL query and both files are incorporated into the catalogue.

Lexicon. As previously described, a *MobilityFNC* expression is used to encode file names that store SQL queries or their corresponding results. In this way, filenames present restrictions based on the operating system. For example, depending on the operating system, some of these characters are not permitted for naming files: *NULL*, \, /, :, %, ?, ∗, ", <, > or |. In compliance with these conditions, *MobilityFNC* does not admit any of these characters. Following the same logic, another restriction imposes a limit of 255 characters per query description, and so the DSL avoids redundancies. Based on these character limitations, the list of possible operators is as follows:

- Principal blocks (source, filter, dimension and operations) are separated using the "+" symbol.
- A new component at the same level is added using the "−" symbol
- A new level and add a component is started using the "_" symbol
- Rows and columns are separated using the "∼" symbol
- A range in the canonical form is defined using the "[␣]" symbol
- A function or method are established using the "{␣}" symbol
- An array of variables are defined using the "[␣,␣]" symbol

In addition to this set of operators, there is a collection of restricted terms to encode queries. Six different classes with an assortment of reserved words are defined as a word-list to formulate queries (see Table 1).

Category	*Terms*
Aggregation	Operations used to calculate (e.g. count, diff, totals, subtotals, top.N or htotal).
Attributes	Attributes listed in the ER model (Figure 3).
Boolean	Any boolean operator defined over the attribute (e.g. pop.over, pop.less, pop.between, pop.equal or applied to other associated attributes).
Ranges	The *-ly* termination referred to a known ranging (e.g. monthly, yearly, etc.), or predefined ones (e.g. summerly or nonsummerly).
Sources	ATM smart cards and TP (single tickets).
Spatial	*coastal*, *municipality*, *land use*, but it could be extended adding more spatial layers.

Table 1: List of *MobilityFNC* terms adapted from the ATMCdT case study.

Syntax and semantics. Once the structure and terms of the DSL are defined, Grammar 1 shows how these elements are combined. This grammar is described using the extended Backus-Naur notation [24] and is also summarised in a more visual manner in Figure 4.

The query definition following this DSL is composed of five blocks that are chained using sum symbols when they describe the structure and contents of the query result or a dot preceding the output file extension. The first block specifies the source(s) of interest to query. In the ATMCdT case, only two different sources are considered: *ATM cards* and *TP single-ride tickets*. The second block contains the filters to apply. There are different kinds of filters, the most important being those used to filter the data by date, time, fare type, or some spatial filters (depending on the spatial layers included in the scenario). In the third block, the dimensions of the query result are defined (rows and columns) including attributes of the tables or derived aggregates. The fourth block includes the aggregation operations performed to achieve the resulting table. The final part of the syntax is the extension of the desired output data format. Until now, only two extensions have been used for storing the query results: **.csv* when the result is a regular table and **.geocsv* when the result contains a spatial column. However, any other extension compatible with this logic could be applied (such as: xlsx, json, geojson, shp, and gpkg).

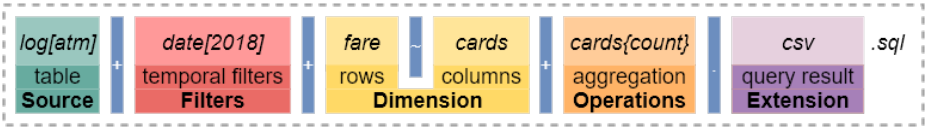

Fig. 4: Example of a query filename encoded with *MobilityFNC*.

Grammar 1 *MobilityFNC* grammar described using the extended Backus-Naur notation [24].

$Filenane ::= Source\_list, " + ", [(Filters, " + ")], Dimension, [(" + ", Operations\_list)], Extension$

$Source\_list ::= "log[", Sources, "]"$
$Sources ::= source|(source, ", ", source)$

$Filters ::= Filter|(Filter, " - ", Filter)$
$Filter ::= ("date[", TempCardTypes, "]")|("cards[", TempCardTypes, "]")|$
$("municipality[", DemSpatial, "]")$
$TempCardTypes ::= ranges|(ranges, ", ", ranges)$
$DemSpatial ::= DemSpatialType|(DemSpatialType, ", ", DemSpatialType)$
$DemSpatialType ::= (spatial)|(boolean)$

$Dimension ::= Rows, " ", Colums$
$Rows ::= (Row, " - ", Row)|Row$
$Row ::= (ComponentType, "[-", atributeFeature, "]")|ComponentType|Filter$
$Columns ::= (Column, " - ", Column)|Column$
$Column ::= (ComponentType, "[-", atributeFeature, "]")|ComponentType|Filter$
$ComponentType ::= attribute|aggregation$

$Operations\_list ::= attribute, "\{", Operations, "\}"$
$Operations ::= aggregation|(aggregation, ", ", aggregation)$

$Extension ::= [(ExtensionStoreFile)]".sql"$
$ExtensionStoreFile ::= ".csv"|".geocsv"$

A workflow for using *MobilityFNC*. As noted earlier, *MobilityFNC* can be applied in different mobility scenarios. In this research work, the DSL is a bridge that enables semi-automating the process for creating new queries. Figure 5 shows an activity diagram that complements the previous sequence diagram (see Figure 2). This activity diagram shows the whole workflow and includes the management of the code (SQL queries), and the repository of results (spreadsheets and GIS files). The activity diagram shows three different roles. The roles of the two researchers (mobility and database manager) have very specific activities: coding filenames using *MobilityFNC* and writing SQL queries respectively; while the repository holds the updated results. This task is automated using a GNU Make program (Makefile). In this workflow, for each query executed on the database, the system automatically creates the corresponding results file, which simplifies the work of the researchers. The central line that corresponds to the outputs repository encompasses all the tasks that can be easily automated (for example, similarity searches and saving or exporting files). Other tasks such as writing the SQL from a textual query are more complex.

3 Writing, naming, and sharing SQL queries

The *MobilityFNC* language has been widely used throughout the project defined in Section 1. Currently, more than 56 queries have been created to analyse various metrics across the bus transport system in the context of a previous research [10]. In this subsection, three different queries are shown and followed for the proposed

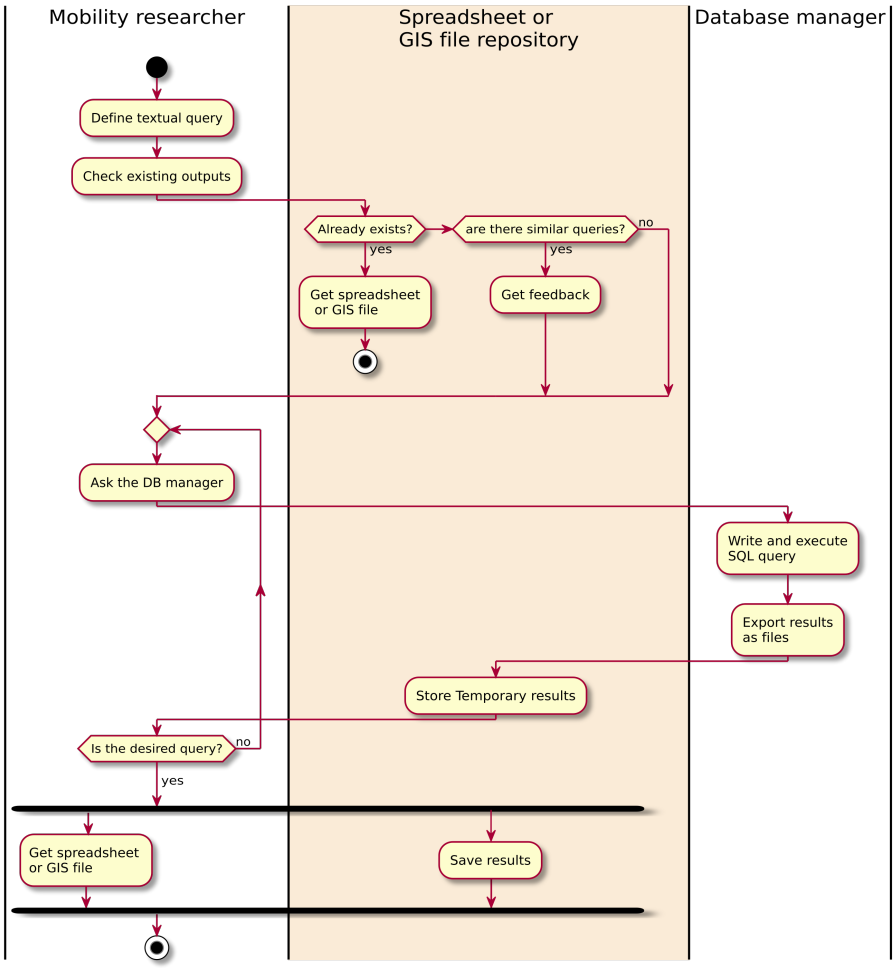

Fig. 5: Activity diagram showing a workflow. *MobilityFNC* is used when the user checks if the query exists and when the results are saved in the repository.

workflow. These queries provide the answers to various research questions by aggregating data and providing spatial context when necessary.

3.1 Do tourists prefer a type of fare?

A first simple example of a *MobilityFNC* query asks about which fares are most used by tourists. More specifically, the query counts the ATMCdT smart cards that were only used in the summer of 2018 (grouped by fare). These cards were active in 2018, and their activity was concentrated in a three-month period. When considering the specifics of each fare type, only the T-10 card (multi-

personal, 10 to 30 transactions, and no expiry date) seems to be the right choice for short stay tourists. There are other fares that are used only in summer, but those fares are intended for longer activity periods and have a higher unitary price per trip. This simple statistic could be interesting for proposing optimisation measures and policies (i.e. better information campaigns). In addition, this query can be used to start a study about those journeys that are concentrated in the summer season. From top to bottom, Figure 6 shows the different stages of the workflow: (1) the query requested by a mobility researcher; (2) the filename structured according to the *MobilityFNC*; (3) its SQL definition based on the proposed database schema; and (4) the associated query result as a table (*.csv).

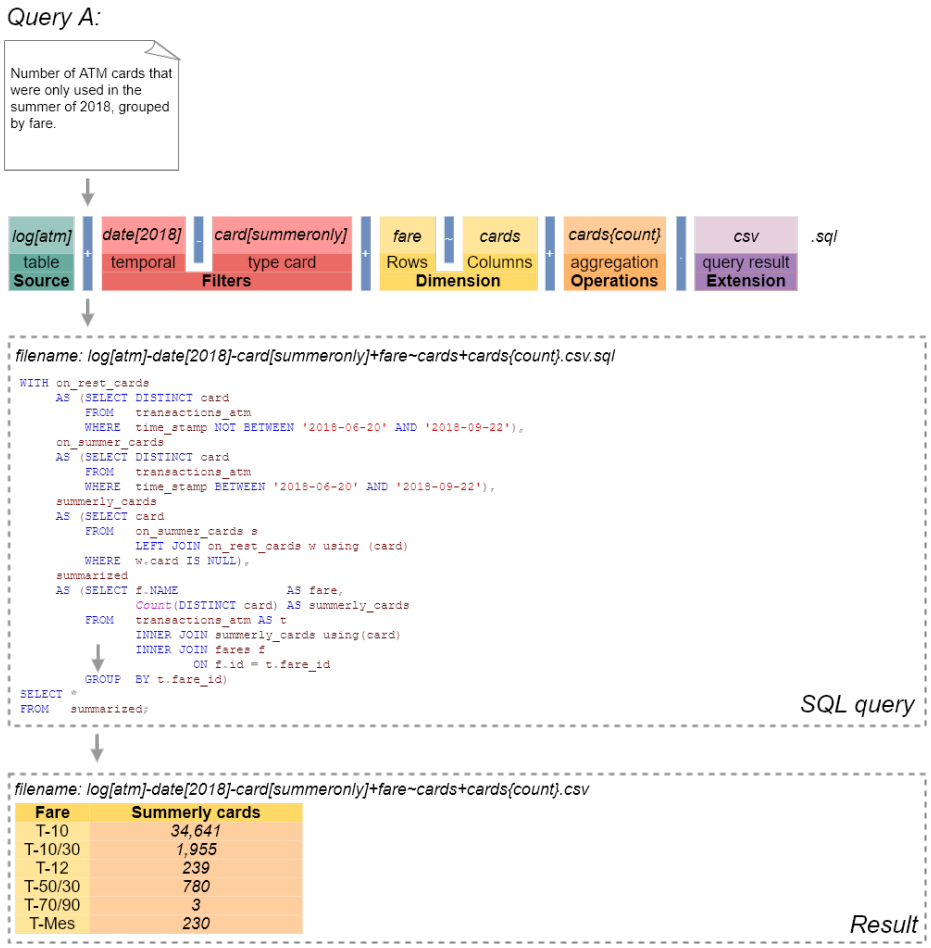

Fig. 6: Example of a query filename encoded with *MobilityFNC*.

3.2 Sale of single-trip tickets in the most touristic cities

A second more complex example is establishing in which period the single-ride tickets are most used in the most touristic cities during one year. The query counts the number of TP transactions (single-trip tickets) that were used in 2018, distinguishing if they took place in summer or during the rest of the year, and only in the most touristic municipalities of the study area (Cambrils, Tarragona, Salou, Reus, and Vila-Seca). The query result shows that the number of tickets sold is more stable in the larger cities (Tarragona and Reus) than in the other three touristic destinations (Cambrils, Salou, and Vila-seca). In these three municipalities, the number of tickets sold in the summer season exceeds those sold in the other nine months of the year. Furthermore, the number of single-trip tickets sold in medium-sized cities such as Cambrils or Salou is similar to or exceeds the number of tickets sold in Tarragona, which is almost four times larger than these others. These figures help to explain the high level of pressure that tourism exerts on public services in the area.

The structure of Figure 7 is the same as in the previous example, showing how the query is computed and the resulting table. In this case, Boolean filters are used in the *dimension block* to avoid redundancy as municipality could also appear in the *filters block*. In this query, a list of totals is calculated to obtain the final count of single-ride transactions per city.

3.3 What is the difference in the use of public transport between a summer and a winter week?

This final query differs from the previous examples in that it takes advantage of the previously described feature of providing a result with a geospatial component as output (Figure 8). A more specific temporal filter is also applied.

This query establishes a comparison to establish the spatial distribution of the greatest pressures on the public transport network, and for this task, only two representative weeks of each season are compared: a summer week and a winter week. In this way, it is possible to discern which stops are most stressed during the tourist period and which are also used throughout the year. As a result of this query, in addition to the columns with the number of transactions in each week and the difference between them, a column is included with the point geometry for each stop. This result is stored in a GeoCSV file, and can be easily uploaded for analysis or visualisation using geospatial data processing tools such as QGIS, gvSIG and R.

4 Concluding remarks and future work

In this chapter, a solution to define and store SQL queries in a multidisciplinary research group is presented. The solution consists of a DSL acting as a file naming convention that strives to normalise the communication workflow between researchers in a project analysing public transportation smart card data. The

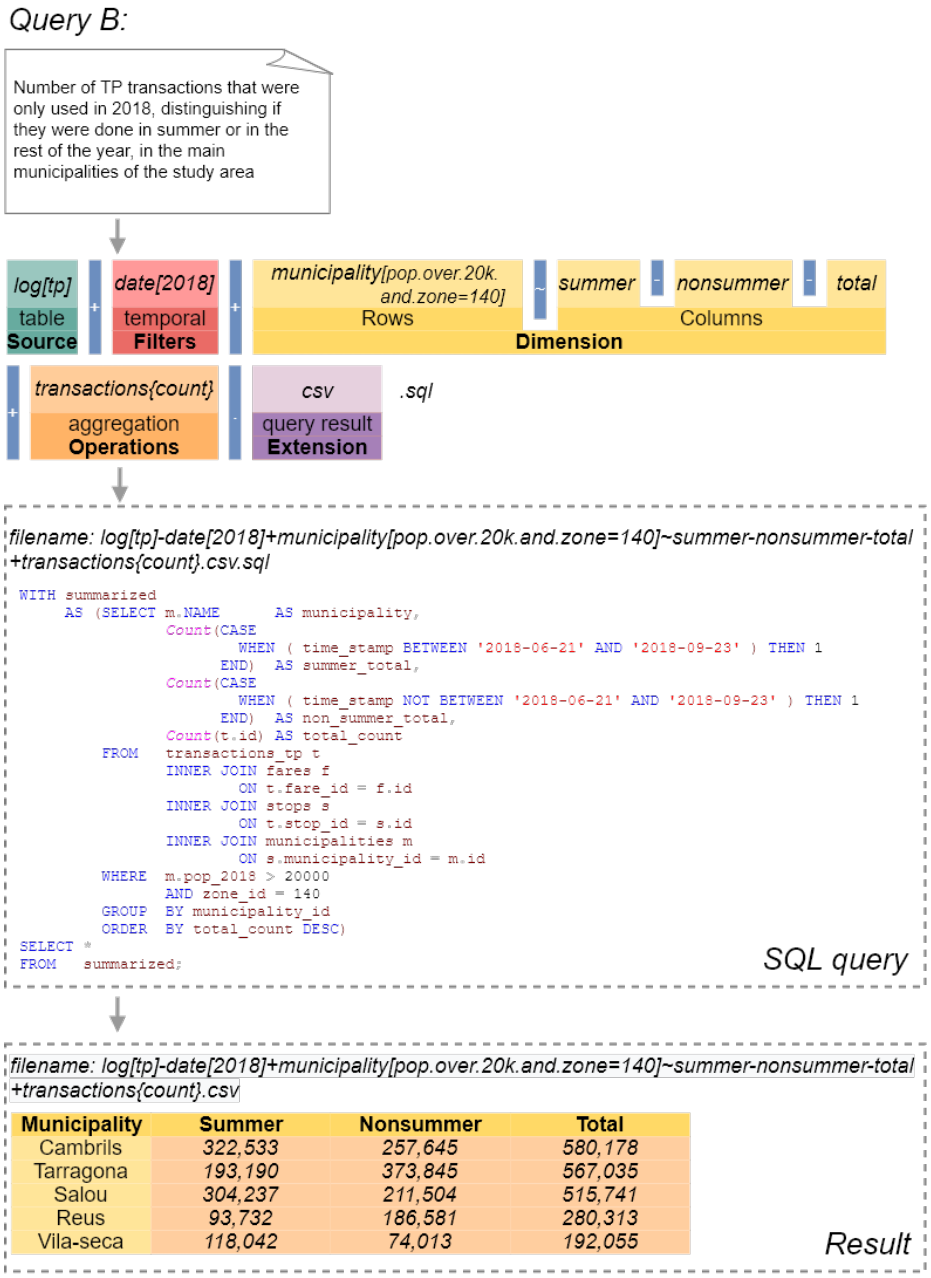

Fig. 7: Example of a query filename with a partial filter, encoded with *MobilityFNC*.

MobilityFNC DSL supports the most common mobility concepts and it can apply temporal and geospatial filters. *MobitityFNC* is designed for mobility experts to be useful in the process of generating SQL queries by database administrators or developers. The SQL query is stored as content in the same file.

A remarkable aspect is the capability to support geospatial outputs in *MobitityFNC* by adding geometries such as point, line, or polygon. This feature can be useful to analyse and visualise the results using GIS software. Currently, this approach has been used in a real project [10], and more than 50 queries have been written, clearly named, and stored without any management issues.

This solution could serve as a bridge to start easily analysing data from other transport consortia and other AFCS, without limitations imposed by software or available resources. The proposed query repository could be publicly shared and the methods reused.

The proposed workflow still needs improvement on some important issues: (1) it is necessary to perform some validation in other projects, including projects studying different types of transport; (2) *MobitityFNC* needs to be used by different multidisciplinary research teams to obtain qualitative and quantitative results as feedback. This implies creating a public code repository for sharing the SQL queries and checking for inconsistencies; (3) and finally, a future improvement would be the development of the capability to automatically generate SQL queries from the *MobilityFNC* format.

References

1. Alsger, A.A., Mesbah, M., Ferreira, L., Safi, H.: Use of Smart Card Fare Data to Estimate Public Transport Origin – Destination Matrix. Transportation Research Record: Journal of the Transportation Research Board **2535**(1), 88–96 (2015). https://doi.org/10.3141/2535-10
2. Bagchi, M., White, P.R.: The potential of public transport smart card data. Transport Policy **12**, 464–474 (2005). https://doi.org/10.1016/j.tranpol.2005.06.008
3. Barth, R.S., Galante, R.: Passenger density and flow analysis and city zones and bus stops classification for public bus service management. In: SBBD. pp. 217–222 (2016)
4. Deursen, A.V., Klint, P.: Little languages: little maintenance? Journal of Software Maintenance: Research and Practice **10**(2), 75–92 (1998)
5. Domènech, A., Gutiérrez, A.: A GIS-Based Evaluation of the Effectiveness and Spatial Coverage of Public Transport Networks in Tourist Destinations. ISPRS International Journal of Geo-Information **6**(3), 83 (mar 2017). https://doi.org/10.3390/ijgi6030083, http://www.mdpi.com/2220-9964/6/3/83
6. Domènech, A., Miravet, D., Gutiérrez, A.: Mining bus travel card data for analysing mobilities in tourist regions. Journal of Maps **16**(1), 40–49 (jan 2020). https://doi.org/10.1080/17445647.2019.1709578, https://www.tandfonline.com/doi/full/10.1080/17445647.2019.1709578
7. Fabbiani, E., Vidal, P., Massobrio, R., Nesmachnow, S.: Distributed big data analysis for mobility estimation in intelligent transportation systems. In: Latin American High Performance Computing Conference. pp. 146–160. Springer (2016)

8. Gokasar, I., Simsek, K.: Using "big data" for analysis and improvement of public transportation systems in istanbul. In: Ase Bigdata/Socialcom/cybersecurity Conference, Stanford University, May 27-31, 2014. Academy of Science and Engineering (ASE), USA,© ASE 2014 (2014)
9. Gokasar, I., Simsek, K., Ozbay, K.: Using big data of automated fare collection system for analysis and improvement of brt-bus rapid transit line in istanbul. In: 94th Annual Meeting of the Transportation Research Board, Washington, DC (2015)
10. Gutiérrez, A., Domènech, A., Zaragozí, B., Miravet, D.: Profiling tourists' use of public transport through smart travel card data. Journal of Transport Geography **88**, 102820 (2020)
11. Gutiérrez, A., Miravet, D.: Estacionalidad turística y dinámicas metropolitanas: un análisis a partir de la movilidad en transporte público en el Camp de Tarragona. Revista de geografía Norte Grande **89**(65), 65–89 (2016). https://doi.org/10.4067/s0718-34022016000300004
12. Gutiérrez, A., Miravet, D.: The determinants of tourist use of public transport at the destination. Sustainability (Switzerland) **8**(9), 1–16 (2016). https://doi.org/10.3390/su8090908
13. Hudak, P.: Domain-specific languages. Handbook of programming languages **3**(39-60), 21 (1997)
14. Idescat: Institut d'estadística de catalunya. Web de l'estadística oficial de Catalunya (2019)
15. Kurauchi, F., Schmöcker, J.D. (eds.): Public Transport Planning with Smart Card Data. CRC Press, 1 edn. (feb 2017). https://doi.org/10.1201/9781315370408, https://www.taylorfrancis.com/books/9781498726597
16. Li, T., Sun, D., Jing, P., Yang, K.: Smart card data mining of public transport destination: A literature review. Information **9**(1), 18 (2018)
17. Lu, Y., Mateo-Babiano, I., Sorupia, E.: Who uses smart card? Understanding public transport payment preference in developing contexts, a case study of Manila's LRT-1. IATSS Research **43**(1), 60–68 (2019). https://doi.org/10.1016/j.iatssr.2018.09.001, https://doi.org/10.1016/j.iatssr.2018.09.001
18. Ma, X., Wu, Y.j., Wang, Y., Chen, F., Liu, J.: Mining smart card data for transit riders ' travel patterns. Transportation Research Part C **36**, 1–12 (2013). https://doi.org/10.1016/j.trc.2013.07.010, http://dx.doi.org/10.1016/j.trc.2013.07.010
19. Morency, C., Trépanier, M., Agard, B.: Measuring transit use variability with smart-card data. Transport Policy **14**(3), 193–203 (2007). https://doi.org/10.1016/j.tranpol.2007.01.001
20. Pelletier, M.P., Trépanier, M., Morency, C.: Smart card data use in public transit: A literature review. Transportation Research Part C: Emerging Technologies **19**(4), 557–568 (2011). https://doi.org/10.1016/j.trc.2010.12.003, http://dx.doi.org/10.1016/j.trc.2010.12.003
21. Prakasa, B., Putra, D.W., Kusumawardani, S.S., Widhiyanto, B.T.Y., Habibie, F., et al.: Big data analytic for estimation of origin-destination matrix in bus rapid transit system. In: 2017 3rd International Conference on Science and Technology-Computer (ICST). pp. 165–170. IEEE (2017)
22. Tao, S., Corcoran, J., Mateo-Babiano, I., Rohde, D.: Exploring brt passenger travel behaviour using big data. Applied geography **53**, 90–104 (2014)
23. Van Deursen, A., Klint, P., Visser, J.: Domain-specific languages: An annotated bibliography. ACM Sigplan Notices **35**(6), 26–36 (2000)

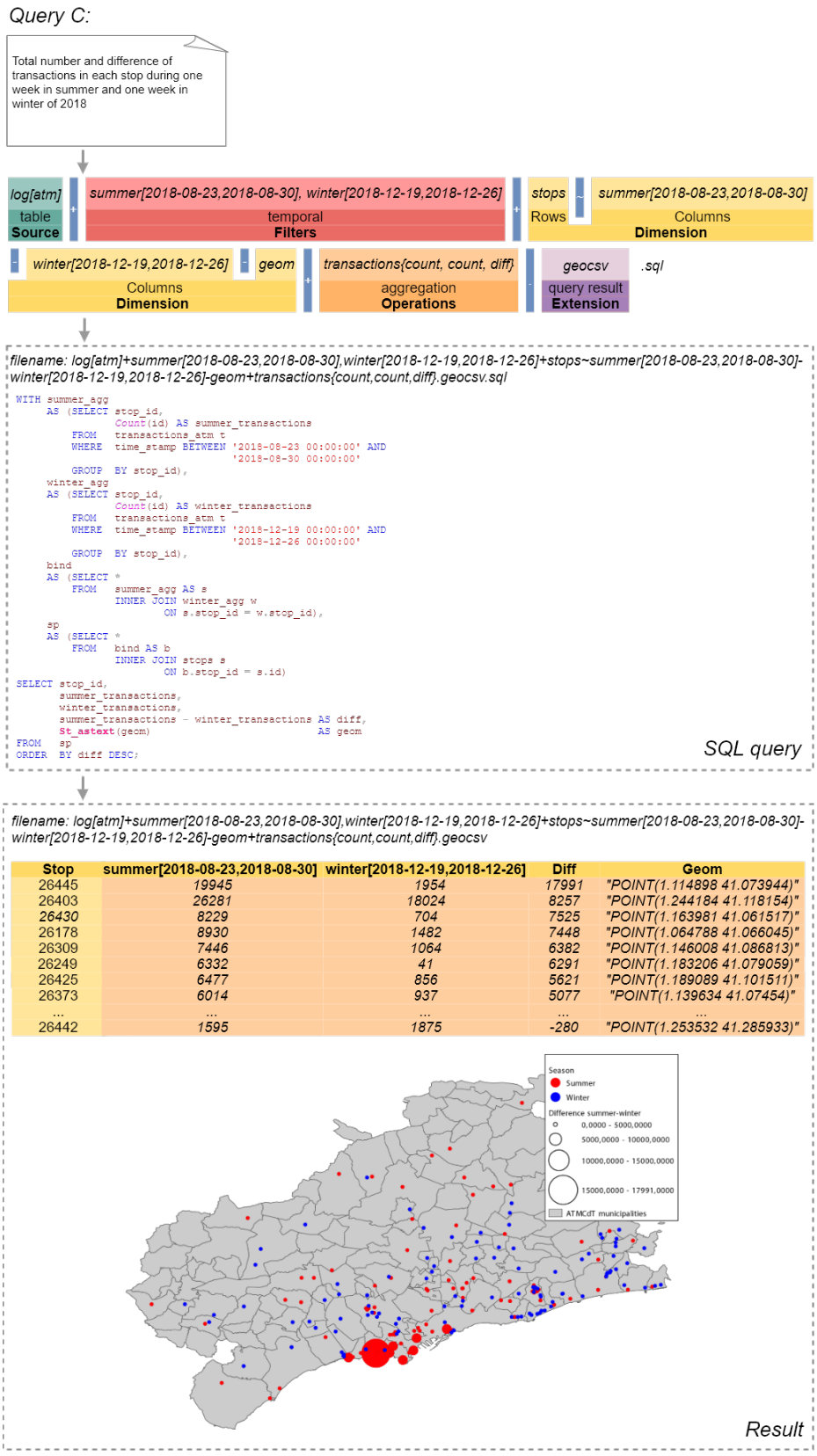

Fig. 8: Example of a query filename with a geospatial output, encoded with *MobilityFNC*.