# QUERYING A TRAJECTORIES DATABASE ABOUT SEX OFFENDERS

Paolino Di Felice, Luca Di Lonardo, and Maurizio Petrocco

Dipartimento di Ingegneria Industriale, Informazione ed Economia,
University of L'Aquila
Via Giovanni Gronchi 18, Nucleo Industriale di Pile, L'Aquila, Italy
paolino.difelice@univaq.it, lucadilonardo86@gmail.com,
maurix86@gmail.com

**ABSTRACT**

*To combat against serial Sex Offenders (SOs), recent laws call for the use of the Global Positioning Satellite (GPS) technology to monitor their movements 24 hours a day. In this paper, we suggest the use of a spatio-temporal database suitable to store the trajectories of such a category of criminals, besides traditional data pertinent to the context (i.e., data about their home, their past history of offences, (pending) crimes, and sensible areas). To show the effectiveness of such a kind of database, then we focus on an investigative strategy that takes profit from the availability of the trips of the SOs in the database and implement it in terms of simple queries. As software platform, the SECONDO DataBase Management System (DBMS) is adopted, since it offers a data type about moving objects as well as a reach set of spatio-temporal operators that greatly simplify the management of trajectories. The solution recommended in our paper opens the frontier to a new generation of software applications, much more effective than those currently used by several criminal investigation departments all over the world, because it keeps together the two components that fully describe any criminal event, namely: space and time.*

**KEYWORDS**

*Sex offender, Criminal mobility, Moving points, Trajectories database, Data analysis, Investigative strategies, Querying.*

## 1. INTRODUCTION

Sex related crimes are a serious social problem because they cause devastating consequences for victims, families, and communities, and because of the high recidivism rate. For instance, a research by the Ministry of Justice of Japan reveals that 30% of serial offenders were responsible for 60% of the crime committed in Japan from 1948 to 2006, [1]. The prevention of recidivism needs effective strategies to supervise SOs. Many jurisdictions have implemented residency restrictions. As quoted in [2], unfortunately residency restrictions have had little measurable crime reducing impact. Another measure against SOs is the so-called "registration and notification policy" that exists in many countries around the world since a long time. However, even this policy did not reduce crime rates tangibly, [3].

More recent laws call for the use of the GPS technology to monitor 24 hours a day the whereabouts of habitual SOs (e.g., [4, 5]). A survey indicated that 34 states in the USA were using GPS for SOs [6]. Moreover, at least six USA's states (California, Florida, Ohio, Missouri, Oklahoma, and Wisconsin) require lifetime electronic monitoring for certain offenders [7]. These acts are the answer to the widespread social expectations, as it can be read in [8] which ends by reporting that "Magistrates and health professionals complain the lack of effective monitoring procedures for released offenders."

DOI: 10.5121/ijdms.2013.5102 

At the web page http://www.nij.gov/nij/topics/technology/maps/welcome.htm (Mapping and Analysis for Public Safety) of the National Institute of Justice (i.e., the research, development and evaluation agency of the United States Department of Justice) we can read the following:

> "Geography has a major influence on crime. … Combining geographic data with police report data and then displaying the information on a map is an effective way to analyze *where*, *how* and *why* crime occurs."

These words sum up correctly the way the software tools, currently being used by the police agencies to carry out investigations, work (useful information on the subject can be found, for example, in [9] and [10]). In other words, it comes up that these software take into account (if any) the temporal component (*when*), that together with the spatial one is indispensable for reaching a full characterization of any criminal event, independently of the spatial component. The need to treat jointly space and time has already been underlined in previous studies pertinent to the crime. A good example in this regard concerned the identification of geographic areas of elevated criminal activity, [11]. In the Introduction of this paper the authors write:

> "The majority of previous hot-spot analyses treat space and time as separate components of crimes, largely ignoring the interaction of space and time in the production of criminal opportunities."

Most of the application software about criminals today in use (e.g., [12, 13]) are fed by data coming from traditional databases (i.e., databases that collect data modeled as standard data types – **`string`**, **`int`**, **`float`**, **`date`**, …) possibly spatially extended (i.e., hosting geometric attributes of type **`point`**, **`line`** and **`region`**, essential to describe, for example, the geographic location of the criminals' home as well as the location where sex offences took place, besides streets, sensible areas, and so on). The already mentioned advent of laws that authorize the monitoring of SOs through the GPS method reopens the discussion about what is the best DBMS technological solution to treat the data about their movements with the final objective of keeping the complexity of the application software as easy as possible. Below, we elaborate a bit on this point.

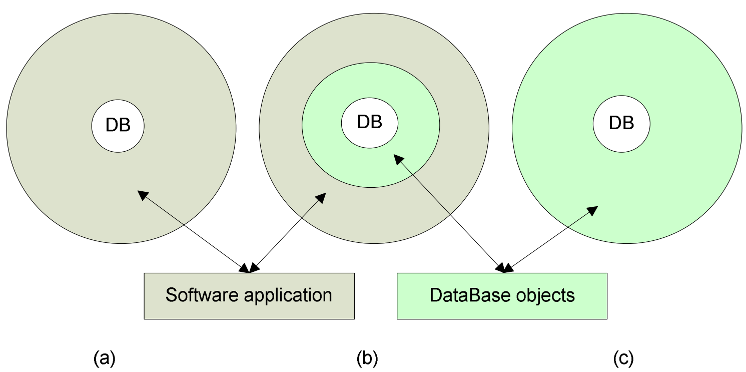


Figure 1. Three possible architecture of software applications on persistent data

Fig. 1a shows the common architecture of a software application on top of persistent data. It highlights that the software interacts *directly* with the database from which it takes data and, if necessary, it writes data in there. The variant of Fig. 1b shows that the application can also interact *indirectly* with the database invoking the execution of some objects of this latter (usually it has to do with views and/or User Defined Functions). Finally, the variant of Fig. 1c concerns the limit case in which the investigations about the SOs are carried out at the working console of the DBMS, calling the execution of some of the database objects. In summary, what it is meant to be conveyed to the reader through Fig. 1 is that any software devoted to carry out investigations about SOs starting from persistent data, can be gradually got lighter until it disappears (clearly a limit case!) as long as, simultaneously, it is increased the number of database objects. With different words, we can say that the two extreme cases of Fig. 1 correspond to two complementary approaches to the development of applications: in the first approach, the development of the software takes place entirely *outside* the DBMS (using an advanced programming language with embedded SQL), while in the second approach the programming takes place entirely *inside* the DBMS (through the use of SQL and, when necessary, by making recourse to the proprietary language of the DBMS itself). An example may help to further clarify the issue. Let us refer to a database about SOs and crimes ascribed to them, made up by the following two tables:

```
criminal   (IDCriminal: string, Name: string, Home: point);

perpetrate (IDCriminal: string, IDCrime: string, When: instant,
            Where: point, Against: string);
```

Let us consider the following two SQL views:

```
CREATE VIEW One (Name, IDCrime, QTY) AS
     SELECT      Name, IDCrime, COUNT(*) AS QTY
     FROM        criminal AS c, perpetrate AS p
     WHERE       c.IDCriminal = p.IDCriminal
     GROUP BY    Name, IDCrime

CREATE VIEW Two (Name, IDCrime, QTY) AS
     SELECT      Name, IDCrime, COUNT(*) AS QTY
     FROM        criminal AS c, perpetrate AS p
     WHERE       c.IDCriminal = p.IDCriminal
     GROUP BY    Name, IDCrime
     ORDER       Name
```

Both views return, for each SO in the database, the number of crimes committed by him for each distinct IDCrime. Two, in addition, sorts the tuples with respect to the field Name. The syntactic differences between the two views are minimal, but from the practical perspective the second view is much more effective than the first one because it facilitates the reading of the results by the end users. Any software application that, after linking the view One, would show the names of the SOs in alphabetical order has to implement a sorting routine in its body or, where available, to invoke one being part of a supporting library. The question becomes thorny when we refer to the context of SOs under GPS surveillance 24 hours a day. In fact, to process the trajectories of their movements, because at the moment there are no libraries of supporting routines, we have to embed into the software application such a complex task, unless we make recourse, once again, to the writing of ad hoc database objects. In this second eventuality, however, it is to be noted that we have an easy life only whether we make use a suitable DBMS, otherwise the problem will be transferred from the outside of the DBMS to the inside. [14] discusses in detail how to deal with this latter situation in the eventuality that we refer to the PostgreSQL/PostGIS open source DBMS.

Since the trajectories are a category of data whose complexity of management goes far beyond the capabilities of current DBMSs, in the present paper it is argued that the time is ripe to adopt a DBMS of a new generation able to support data types and operators for the so-called *moving points* (in short, *m-points*), as meant in [15]. M-points are time dependent geometries. Two views on m-point data have been established in the past. The first one focuses on answering questions on the current position of m-points, and on their predicted temporal evolution in the (near) future. This approach is often called *tracking* [16]. A second approach represents (in a single database attribute) complete histories of m-points inside so-called *trajectory databases* [15]. Of the two approaches, the second is better suited, in our opinion, to be used to study the movements of SOs mitigating, at the same time, the issue of privacy invasion even more pronounced in the case of tracking that implements a real-time control of people actions.

Choice, the latter, that although decidedly "punitive" is not supported by experimental studies that prove it enhances people safety. In case, so far, it is true the opposite. For instance, Payne & DeMichele, [2], report that there is a large agreement among the domain experts that GPS-based security policies provide “a false sense of security”, since there are no evidences that the electronic monitoring in itself reduces recidivism.

A further important motivation in favor of the adoption of a database of trajectories comes from the consideration that this strategy is the only one applicable in those cases where the jurisdiction implements the so called *passive reporting method* (the offender's movements are uploaded in a monitoring center repository usually once a day), [17].

Our paper has the following merits:

a) it intercepts the request to subject the serial SOs to continuous monitoring via GPS posed by recent regulations;
b) it proposes and implements a solution that integrates the two components present in each criminal event, i.e., the space and the time. To separate time from space is quite impossible. In fact, the physical environment we inhabit does not only consist of the spatial component. Inseparable from it is the temporal one, which is just as important, even if it may not display itself in the same tangible manner. Papers like, for example, [18] already stressed the benefits that approaches that combine space and time could bring to check alibis in police investigations. Time geography traces its roots back to the Swedish geographer Hägerstrand, [19]. The solutions implemented so far in the software tools available to the investigators to carry out their investigations, on the contrary, are space-oriented, that is, they pose the greatest emphasis on the spatial distribution of crimes and only after involve (when this happens) the time component;
c) it designs a “nucleus” of a spatio-temporal database about the history of the movements of SOs besides “usual data” for this kind of context. Putting into operation such a kind of database is the preliminary step to be carried out if we want to speed-up the implementation of any investigative strategy against serial SOs;
d) it implements the designed database by adopting the SECONDO DBMS, an innovative software thought to handle natively moving objects. SECONDO is an open source software at an advanced stage of implementation at the FernUniversität in Hagen, [20];
e) it reaffirms, in continuity with the software solutions already available (e.g., [12, 13]), the centrality of the adoption of the metaphor of the maps to return the results of the analysis to the end-users in a way that is easily comprehensible by them.

The remainder of the paper is structured as follows. Sec. 2 introduces the reference scenario about SOs and their movements. Then, it introduces both a spatio-temporal database collecting the data about the problem (i.e., (potential) crimes, sensible areas, subjects on probation, their past crimes, and their trips) and its implementation in the SECONDO DBMS. Lastly, it describes an example dataset used to feed the database. Sec. 3 focuses on an investigative strategy that takes profit from the availability in the database of the trips of the SOs, showing its implementation in SECONDO, as well. Sec. 4 concludes the paper.

## 2. SEX OFFENDERS: THE APPLICATION CONTEXT, A MODELING DATABASE, AN EXAMPLE DATASET

### 1.1. The Application Context

As already mentioned, the reference scenario of the study summarized in this paper concerns sensible areas (schools, parks, public restrooms, train stations, ...) and SOs whose movements between sensible areas are tracked (hereinafter, therefore, also called *subjects on probation*). The position of the SOs is acquired by an electronic device equipped with a GPS detector. Such data are transmitted to a server and filed for a certain number of years (according to the national and international laws in force). Those GPS tracks are supposed to be read only when any crimes are reported, to determine if any offenders were nearby the place of the offence. The application context we refer to looks like that of many police departments scattered around the world. This is the case, for example, of the Sexual Predator and Offender Tracking Unit of the Pinellas County (Florida) mentioned in [21] . For each SO, the "history" of the past offences for which him has been arrested is also taken into account, besides the list of unsolved crimes.

In the remainder of the paper we use the following notations:

| | |
|---|---|
| $\mathcal{A} = \{A_1, A_2, \ldots, A_a\}$: | the sensible areas, |
| $\mathcal{S} = \{S_1, S_2, \ldots, S_s\}$: | the subjects on probation, |
| $\mathcal{O} = \{O_1, O_2, \ldots, O_s\}$: | the history of past offences of each subject on probation in $\mathcal{S}$, |
| $\mathcal{T} = \{T_1, T_2, \ldots, T_t\}$: | the movements of SOs, |
| $\mathcal{C} = \{C_1, C_2, \ldots, C_c\}$: | the known sexual crimes, |
| $\mathcal{P} = \{P_1, P_2, \ldots, P_p\}$: | the pending sexual crimes. |

In turn, Ti = {<P,t> | *P* is a point described by a pair of coordinates <lat, long> and *t* is the time stamp of the acquisition of *P*}, where i=1, 2, ..., $|\mathcal{T}|$. The elements of $T_i$ are temporally ordered. $T_i$ expresses the trip of a subject in $\mathcal{S}$ between two sensible areas in $\mathcal{A}$.

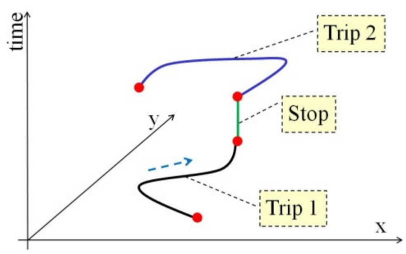


Figure 2. The whole GPS history of a subject on probation made up of intermediary trips

An explicit comment can explain how it is possible to have data about the movements of the subjects on probation from a sensible area to another one. The premise is that the acquisition of the position of the generic subject is permanent. The daily mobility of each subject can be summarized by a set of "single" trips that he performed during the day (Fig. 2). Therefore, this continuous stream of information contains different trips made by the subject. In order to distinguish between them, we need to detect when he stopped for a while in a place. This point in the stream will correspond to the end of a trip and the beginning of the next one.

In this way splitting the "whole history of the movements" of a subject into single trips becomes trivial. It arises, therefore, the need to process the movements of a subject in order to cut them into "elementary" units. For the purposes of this study, we assume that the single trip goes from the exit of a sensible area to the entrance into another one. Commonly, the stop points are identified knowing that they keep the same spatial position for a certain amount of time quantified by a predefined temporal threshold (e.g., [22]). In our study, vice versa, it will be appropriate to adopt a more rough schematization consisting in ignoring the movements that a SO performed inside a sensible area. Therefore, all the time spent within the area is as if he had been stopped.

Another specificity of the reference scenario of our study concerns the fact that the SO did not necessarily enter into the sensible areas. Let us consider, for example, a school with a fence and locked gate during the hours of teaching. Likely, he preferred get close enough to the building and remain under observation in order to spy on potential victims without the need to enter and without risking of being seen. To deal with this aspect, it is sufficient to apply a "buffer" to the polygon that defines the geometry of the building. If the location of the SO falls within the buffer, to the purpose of our study it is as if he was inside the school.

## 1.2. A Modeling Database

The values from the sets $\mathbb{C}$, $\mathbb{P}$, $\mathbb{S}$, $\mathbb{A}$, $\mathbb{T}$, and $\mathbb{O}$ can be stored, in sequence, in a SECONDO spatio-temporal database structured in terms of the following six tables:

```
let crime = [const rel(tuple ([IDCrime: string, Description: string]))
    value ()]
let pendingCrime = [const rel(tuple([IDPC: string, When: instant, Where:
    point, Description: string, Against: string])) value ()]
let criminal = [const rel(tuple ([IDCriminal: string, Name: string,
    Home: point])) value ()]
let sensibleArea = [const rel(tuple ([IDSA: string, Type: string,
    City: string, State: string, Position: point, Layout: region]))
    value ()]
trip (IDCriminal: string, TripData: mpoint, From: string, To: string)
let perpetrate = [const rel(tuple ([IDCriminal: string,
    IDCrime: string, When: instant, Where: point, Against:
    string])) value ()]
```

The **`pendingCrime`** table requires a remark. Such a table contains a tuple for each unsolved crime. During the life of the database, each time a crime is attributed to a SO, then an ad hoc script that migrates the corresponding values of the attribute **`Where`**, **`When`**, and **`Against`** (from **`pendingCrime`**) into a tuple of the **`perpetrate`** table has to be executed; tuple to be completed with the values of the attributes: **`IDCriminal`** and **`IDCrime`**. Immediately afterwards it will be necessary to delete from **`pendingCrime`** the tuple migrated into **`perpetrate`**.

An explicit remark about the attributes **`from`** and **`to`** of table **`trip`** is required, because there are reasons to cancel them and others to keep them. In case the scenario we refer to in this study would be implemented, then the information about the initial and final point of each trip would be possible to be inferred by calculating the intersection between the m-point and the geometry of the sensible areas (hence attributes **`from`** and **`to`** could be canceled). Vice versa, working in the abstract of this section, the movements between pairs of sensible areas of each subject on probation are generated for our purposes, so it is quite reasonable to keep track of them explicitly in the **`trip`** table. There is, however, a further motivation that suggests keeping attributes **`from`** and **`to`** even in the real scenario and it is related to performances. In fact, all the times we will need to know the extreme areas of a trip, it will be sufficient to carry out a query that makes a look-up of the **`trip`** table, vice versa those data have to be computed "from scratch".

Di Felice & Falcone, [23], pointed out that the SECONDO platform overcomes standard ones (i.e., IBM DB2-SE, Oracle/Spatial, PostgreSQL/PostGIS, …) with respect to: the elegance of implementation, the feasibility of spatio-temporal analysis, and the performances.

## 1.3. An Example Dataset

To make preliminary experiments, we refer to a synthetic dataset where the movements of SOs have been generated by a Java program which receives as input the sensible area of departure and arrival, the date and start time of the trip and returns a text file that describes the journey. The example dataset is small, but significant enough for the purposes of the study. It consists of 20 subjects and 20 sensible areas (located in the town of L'Aquila and nearby, centre of Italy – Fig. 3, and composed of 7 elementary schools, 6 middle schools, 4 public restrooms, and 3 parks).

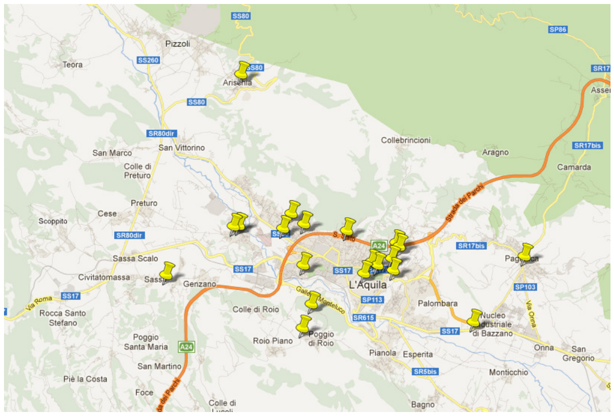


Figure 3a. The Google map about the sensible areas being part of the example dataset

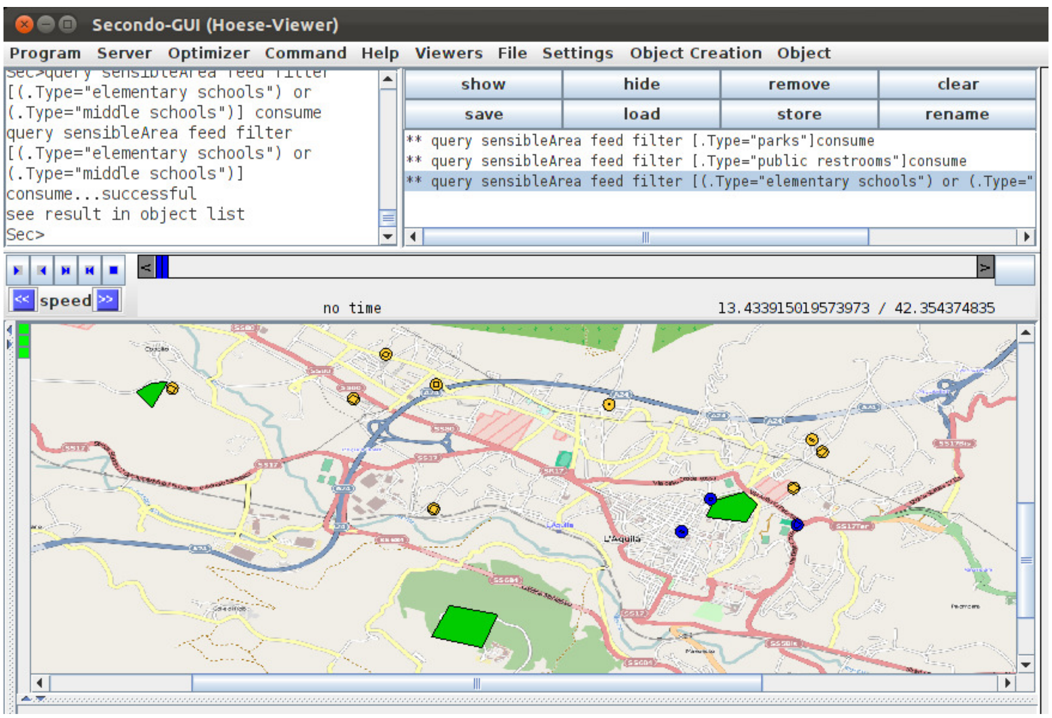


Figure 3b. The map of the sensible areas in the dataset shown by SECONDO. The output is generated by the three queries shown in the top-right window of the figure, according to the selected visual style for the returned spatial objects: green areas represent parks, yellow (blue) circles represent schools (public restrooms)

*Remarks about the query syntax.*

The keyword **query** starts queries written in the SECONDO language. The **feed** operation reads a relation (e.g., **sensibleArea**) from disk and puts its tuples into a stream as they are. The **consume** operation collects a tuple stream into a persistent relation suitable to be kept and indexed. The **filter** operation is equivalent to the **WHERE** clause of SQL as can be understood by looking at the query below that reformulates the first one above: **SELECT * FROM sensibleArea WHERE Type = "parks"**

The database on which the tests were conducted also contains the data of 30 trajectories corresponding to as many trips of the 20 subjects on probation between pairs of sensible areas.

These movements are shown by the graph of Fig. 4, where each node is labeled with the code of a sensible area, while the arcs are labeled with the code of the offender that moved between the extreme nodes.

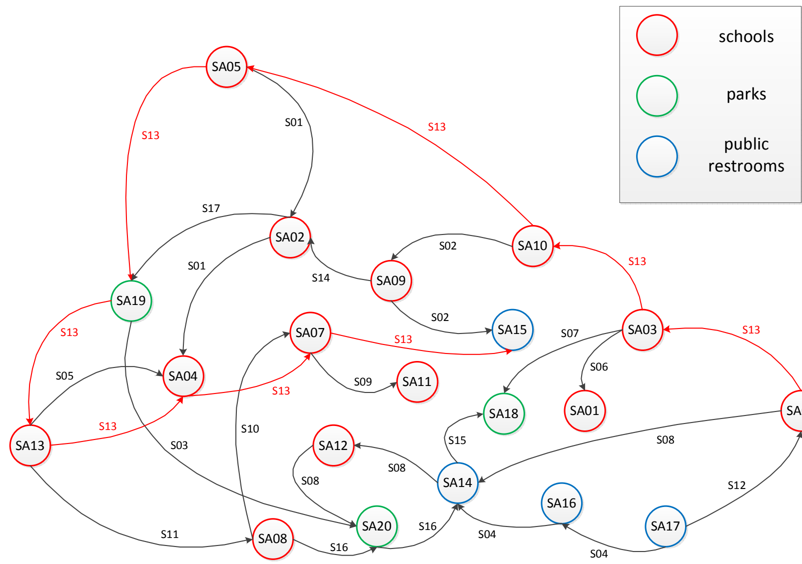


Figure 4. The graph of the movements of the 20 subjects on probation in the example dataset

The reader interested to learn how to load tracking data collected by a GPS device into a SECONDO attribute (**`tripData`** in our case) of a relation (**`trip`**) as a **`mpoint`** data type may refer to [24].

## 3. AN INVESTIGATIVE STRATEGY

At a high level of abstraction, investigators need methods of inquiry that support them trying to:

- limit the list of potential culprits from which to start the investigation downstream of an abuse (in short, *punitive* action),
- prevent the frequent episodes of recidivism among the serial SOs registered in the databases they have access to, through the periodic analysis of their movements (in short, *predictive* action).

The two actions have complementary objectives and should therefore be supported through the use of different software tools. The most effective approaches to support predictive investigations are listed below:

- the first approach is based on data mining methods. [25, 26, 27] treat the topic from different perspectives.
- In the second approach, proposed by Porter & Reich, [28], the objective of the analysis is predicting the location of the next event given a crime series by estimating a conditional spatial density function.
- Lastly, CrimeStat, [12], implements to so-called *Correlated Walk Analysis* to understand the sequencing of events in terms of time interval, distance and direction.

It is pertinent to add that some simple investigations, aimed at identifying predictive spatio-temporal patterns that usually go with sex offences, can be carried out by applying to the trajectories of the SOs stored into the database some of the spatio-temporal operators featured by SECONDO. This is the case, for example, we want to know the length of the stay (measured in minutes) of each criminal inside the sensible areas of a county in a given temporal window (measured in days, weeks or months), so that to be able to carry out selective checks for each "anomalous" stop (for example, the permanence of few hours in a public restroom). In such an eventuality, it will be sufficient to define ad hoc database's objects to be linked from the outside.

In the following, we define and implement an investigative strategy (shortly called the *Test*) belonging to the punitive action category. The goal is to emphasize the strength of the proposed innovative technological solution which is able to process natively the trajectories accomplished by the SOs, a kind of data that goes far beyond the capabilities of current DBMSs. As we shall see shortly, in the framework of the existence of a spatio-temporal database of the type proposed in this article, the implementation of the *Test* is trivial. A further merit of this test is its generality, in fact, it applies both to the case of a single episode of sexual abuse than serial.

Queries can be formulated to SECONDO either in an SQL-like language (top level) or in the executable language (low level). Hereinafter, we use this latter way. As soon as the SECONDO optimizer will be more stable than it is at the present time, it will be the case to move to use SQL.

*The investigation requirement*

The idea is the following. Downstream of an episode of sexual violence, a rapid check to be carried out by querying the spatio-temporal database is to verify, for each subject on probation stored in it, whether he has made a movement that brought him to pass near by the crime location (*where*) the same date of the crime and in an hour close to that of the abuse (*when*). If the *Test* result was positive, it would be a very strong evidence against the subject. This test expresses a need strongly felt by the anti-crime units around the world. An evidence may be found in [21], where we can read:

> "*When any crimes are reported in Pinellas County, the Sexual Predator and Offender Tracking Unit uses GPS tracking to determine if any offenders were nearby*."

Evidently, the SPOT Unit does not use a DBMS like SECONDO, but a "turnkey" software solution that accesses the files about the tracks of the SOs and display them on a screen where it is also shown the place of the crime. Such a solution corresponds to the software architecture of Fig. 1a.

*Input-output of the Test*

INPUT

$\mathbb{S}$, $\mathbb{T}$, *where* (the place of the abuse), *when* (date and time of the abuse)

OUTPUT

$\mathbb{S}_{where}^{when} \subseteq \mathbb{S}$ (*i.e., the list of criminals* for which in the spatio-temporal database is stored a movement that saw them be in proximity of *where* in a time stamp close to *when* – i.e., same date of the abuse, and time close).

*Implementation*

It is accomplished in two steps discussed below in sequence.

a) Build a buffer around the crime site. It is necessary to have recourse to a buffer to absorb all the causes of uncertainty about the exact location of the criminal (alias the m-point) [29]. The buffer radius is sufficient to be of some tens of meters:

```
let buffer =
  pendingCrime feed extend [Area:circle (.Where, 0.0001, 50)]
consume
```

b) Process the movements of each SO in the spatio-temporal database which began before the abuse and ended just after, in order to assess whether he entered the area covered by the buffer in a moment close to that of the crime.

```
query trip feed {tr} buffer feed {bf}
```

```
  symmjoin[.TripData_tr passes ..Area_bf]
  extend [Range:theRange(inst(initial(.TripData_tr at .Area_bf)),
          inst(final(.TripData_tr at .Area_bf)), FALSE, TRUE),
          Trajectory: trajectory(.TripData_tr)]
  filter [.Range intersects theRange(
          .When_bf - [const duration value (0 3600000)],
          .When_bf + [const duration value (0 3600000)], FALSE, TRUE)
         ]
  renameattr [IDCriminal:IDCriminal_tr, IDPC:IDPC_bf,
          Area:Area_bf, When:When_bf]
  project [IDCriminal, IDPC, Area, When, Trajectory]
consume
```

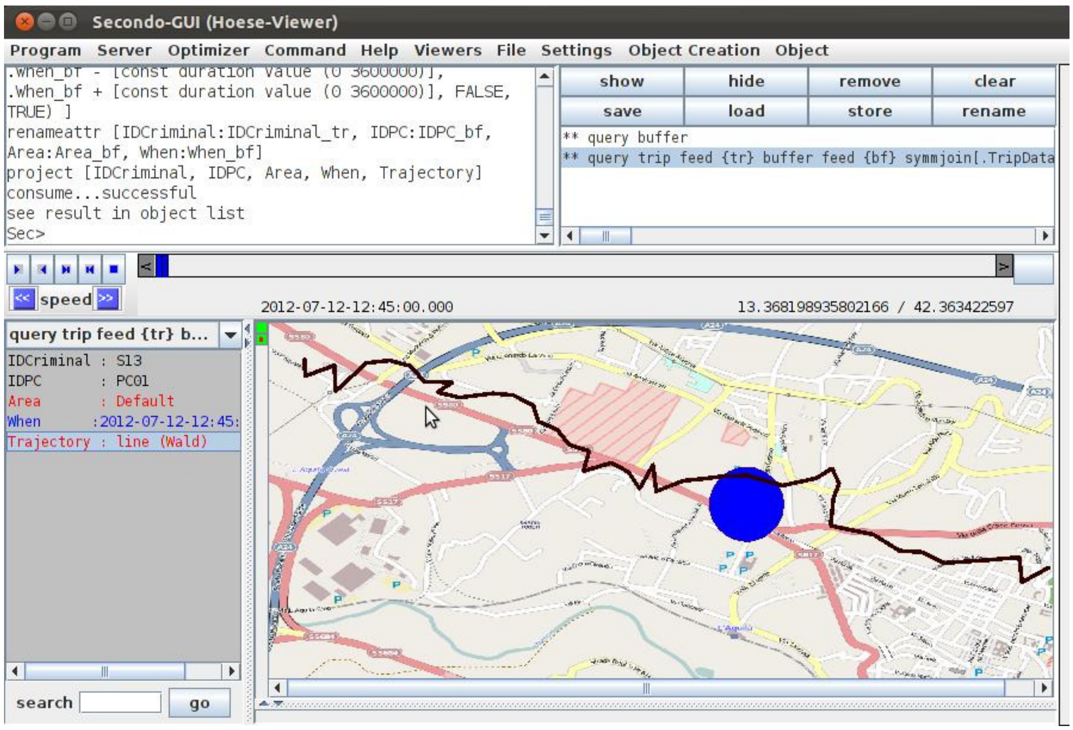


Figure 5. The output of the *Test*. *Default* and *Wald* are SECONDO's categories of visualization. The trajectory of the SO does not match the road network because the trips of the criminals have been generated by a Java program. However, this does not limit in any way the generality of the two queries that implement the *Test*

The operator **`passes`** ascertains which of the subjects on probation registered in the spatio-temporal database entered the computed buffer area and, for each of them, the time of entry and exit from such an area is calculated (operators **`inst(initial())`** and **`inst(final())`** applied to the m-point (attribute **`tripData`**)). Then, **`theRange()`** builds the corresponding time window. To see the path of the SO, the output of the query is extended with the attribute **`Trajectory`** that contains the trajectory of the m-point, calculated by applying the operator **`trajectory`** to the field **`tripData`** of type **`mpoint`**. Through the unary operator **`filter`** which, in turn, uses the **`intersect`** operator, the suspects are shrank to only the SOs who have passed through the buffer in a temporal window between one hour before and one hour after that of the abuse (*When*).

Fig. 5 shows the result of the *Test* applied to the example spatio-temporal database; i.e., there is only one trajectory (ascribable to the subject on probation *S13*) that crosses the buffer (**`Area`**) built around the crime site (*Where*), this latter identified by code IDPC = "PC01".

## 4. CONCLUSIONS AND FUTURE WORK

The advent of recent laws, that authorize the monitoring 24 hours a day of SOs through the GPS method, reopened the discussion about what is the best DBMS technology to manage their movements. The issue is relevant because the more effective is the management of the trajectories at the storage level and the easier and faster will be the coding of any software tool that draws from the repository. Our conclusions can be summarized as follows: at the conceptual level, the best organization of the database can be achieved by modeling the subjects' trips as m-points while, at the implementation level, today the SECONDO DBMS is the most mature enabling technology since it supports natively a data type about moving objects (**`mpoint`**) as well as a reach set of spatio-temporal operators to query such complex data. A double facilitation which makes the modeling of the trajectories of the SOs immediate thanks to the **`mpoint`** data type and that speeds up the subsequent querying of the database. Both these aspects were exemplified in the article.

We are convinced that the adoption of a DBMS for moving objects opens the frontier to a new generation of software tools much more effective than those currently used by several criminal investigation departments all over the world because it treats jointly the two components which are present in each criminal event, that is, space and time. The concrete benefits of this innovation will fall on the entire community that will have the chance to live in safer cities.

### 4.1. Future Work

Once a database like that designed in this paper will be put in exercise, then time should be spent to understand the many implications deriving from the availability of a type of data much richer than that ever used before: the trajectories of SOs over long time periods. In particular, new investigative algorithms, that take benefit from the knowledge of the full history of the movements of the SOs as well as of their past history (places of sexual abuses and modus operandi), should be investigated. The *Test* discussed in the paper is just an elementary example.

The paper touched, indirectly, the question whether in the context of SOs the time is ripe to move from the classical geography metaphor to that of time geography introduced by Torsten Hägerstrand at the end of the sixties. This issue too should be further investigated.

## REFERENCES


[1] Someda, K. (2009). "An international comparative overview on the rehabilitation of offenders and effective measures for the prevention of recidivism", *Legal Medicine*, 11, 82–85.

[2] Payne, B. K., & DeMichele, M. (2011). "Sex offender policies: Considering unanticipated consequences of GPS sex offender monitoring", *Aggression and Violent Behavior*, 16, 177–187.

[3] Lieb, R., & Kemshall, H. T. (2011). "Post-release controls for sex offenders in the U.S. and UK", *International Journal of Law and Psychiatry*, 34, 226–232.

[4] *Adam Walsh Child Protection Act*. (2006). Public Law 109-248. 42.

[5] *Jessica's law*. (2007). "State of California, Department of Corrections and Rehabilitation, Division of Adult Parole Operations", Retrieved Oct. 2012, from http://www.interstatecompact.org/Directory/RegionsStates/West/California.aspx.

[6] Interstate Commission for Adult Offender Supervision. (2007). *ICAOS GPS update survey*, Lexington, KY: Author.

[7] Dunlap, K. (2010). "Sex offenders after prison: Lifetime GPS monitoring?", Retrieved Oct. 2012, from http://blogs.findlaw.com/blotter/2010/02/sex-offenders-after-prison-lifetime-gps-monitoring.html.

[8] Kaplan, L. V. (2008). "Sexual psychopathy, public policy, and the liberal state", *International Journal of Law and Psychiatry*, 31, 172–188.

[9] Kemshall, H. (2001). "Risk Assessment and Management of Known Sexual and Violent Offenders: A review of current issues", *Police Research Series*, Paper 140.

[10] Rich, T., & Shively, M. (2004). *A Methodology for Evaluating Geographic Profiling Software*. Document No. 208993.

[11] Grubesic, T. H., & Mack, E. A. (2008). "Spatio–Temporal Interaction of Urban Crime", *Journal of Quantitative Criminology*, 24, 285–306.

[12] Levine, N. (2012). *CrimeStat III: A Spatial Statistics Program for the Analysis of Crime Incident Locations* (version 3.3). Ned Levine & Associates, Houston, TX.; National Institute of Justice, Washington, DC. July. Program and documentation available at: http://www.icpsr.umich.edu/CrimeStat.

[13] O'Leary, M. (2012). "Improving Mathematical Approaches to Geographic Profiling", *Technical report*. Retrieved Oct. 2012, from http://pages.towson.edu/moleary/Profiler.html.

[14] Di Felice, P. (2012). "A Short Term Solution to Implement Applications about Moving Points on top of Existing DBMSs", *International Journal of Computer Applications* (0975 – 8887) Volume 50 – No.10, July 2012. DOI: 10.5120/7804-0934.

[15] Güting, R. H., & Schneider, M. (2005). *Moving Objects Databases*. Morgan Kaufmann Publishers.

[16] Sistla, A. P., Wolfson, O., Chamberlain, S., & Dao, S. (1997). "Modeling and Querying Moving Objects", *Proc. 13th International Conference on Data Engineering*, 422–432.

[17] DeMichele, M., & Payne, B. (2009). *Offender supervision with electronic monitoring: A user's guide*. Washington, D.C.: Bureau of Justice Assistance, Office of Justice Programs, Department of Justice. Retrieved Oct. 2012, from http://www.appa-net.org/eweb/docs/APPA/pubs/OSET_2.pdf.

[18] Neutens, T., Versichele, M., & Schwanen, T. (2010). "Arranging place and time: A GIS toolkit to assess person-based accessibility of urban opportunities", *Applied Geography*, 30, 561–575.

[19] Hägerstrand, T. (1970). "What about people in regional science?", *Papers of the Regional Science Association*, 24(1), 7–21.

[20] Güting R. H., Behr, T., & Düntgen, C. (2010). "SECONDO: A platform for moving objects database research and for publishing and integrating research implementations", *IEEE Data Engineering Bulletin*, 33(2), 56-63.

[21] Preston, K. (2009). "Right Place, Right Time: GPS Monitoring in Pinellas County*", A Quarterly Bulletin of Applied Geography for the Study of Crime & Public Safety*. 2(1), 3–5.

[22] Trasarti, R., Pinelli, F., Nanni, M., & Giannotti, F. (2012). "Individual mobility profiles: methods and application on vehicle sharing", *20th Italian Symposium on Advanced Databases Systems*.

[23] Di Felice, P., & Falcone, A. (2012). "An Algorithm and a Database: two Conceptual Tools to Control the Diffusion of Animal Diseases", *Journal of Advances in Information Technology*. To appear.

[24] Güting R. H. (2011). Example: Loading Tracks and Displaying Them With a Map Background. (Retrieved on July 2012 from http://dna.fernuni-hagen.de/Secondo.html/files/ SecondoTracks.pdf).

[25] McCue, C., & Parker, A. (2003). "Connecting the dots: Data mining and predictive analytics in law enforcement and intelligence analysis", *Police Chief*, 70,115-122.

[26] McCue, C. (2007). "Data Mining and Predictive Analysis: Intelligence Gathering and Crime Analysis", ISBN-10: 0750677961, ISBN-13: 978-0750677967.

[27] Giannotti, F., Nanni M., Pedreschi, D., Pinelli, F., Renso, C., Rinzivillo, S., & Trasarti, R. (2011). "Unveiling the complexity of human mobility by querying and mining massive trajectory data", *The VLDB Journal*, 20, 695–719 (DOI 10.1007/s00778-011-0244-8).

[28] Porter, M. D. & Reich, B. J. (2012). "Evaluating temporally weighted kernel density methods for predicting the next event location in a series", *Annals of GIS*, 18(3), 225–240. DOI:10.1080/19475683.2012.691904.

[29] Trajcevsky, G., Wolfson, O., Hinrichs, K., & Chamberlain, S. (2004). "Managing uncertainty in moving object databases", *ACM Transactions on Database Systems*, 29(3), 463-587.