Data & Knowledge Engineering 91 (2014) 1–16

Contents lists available at ScienceDirect

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

Editorial

# Geospatial data streams: Formal framework and implementation

Z. Galić [a,*], M. Baranović [a], K. Križanović [a], E. Mešković [b]

[a] University of Zagreb, Faculty of Electrical Engineering and Computing, Department of Applied Computing, Unska 3, 10000 Zagreb, Croatia
[b] University of Tuzla, Faculty of Electrical Engineering, Franjevačka 2, 10000 Tuzla, Bosnia and Herzegovina

ARTICLE INFO

*Article history:*
Received 6 December 2011
Received in revised form 2 February 2014
Accepted 7 February 2014
Available online 22 February 2014

*Keywords:*
Cross-disciplinary applications
Data models
Data streaming
Query
Spatial/temporal data streams

ABSTRACT

A spatio-temporal database manages spatio-temporal objects and supports corresponding query languages. Today, the term moving objects databases is used as a synonym for spatio-temporal databases managing spatial objects with a continuously changing geospatial location and/or extent. Recent advances in wireless communication, miniaturization of spatially enabled devices and global navigation satellite systems (GNSS) services have resulted in a large number of novel application domains. Applications in these novel domains (geo-sensor networks, moving objects tracking, real-time traffic analysis, etc.) process huge volumes of continuous data streams, i.e. data sets that are produced incrementally over time, rather than those available in full before the processing begins. Several data stream management systems (DSMSs) have been developed to manage this data. Since they are mainly based on a relational paradigm, they do not support geospatial data. Therefore, there is an urgent need for geospatial data stream management, ranging from real-time monitoring and alerting to long-term analysis of processed geospatial data. In this paper we present a formal framework consisting of data types and operations needed to support geospatial data in data streams. It can be used as a basis either for implementation of a completely new geospatial DSMS, or for extending available open source products and research prototypes. We leverage the work on abstract data types from spatio-temporal databases, present an implementation based on user-defined aggregate functions and illustrate embedding into an SQL-like language.



## 1. Introduction

Over the years DBMSs were enhanced in order to better accommodate various domains of application. Hence, a modern DBMS can support geospatial, multimedia, temporal, XML, streaming and other types of data. In the last two decades, extending DBMSs in the direction of managing moving objects data and data streams has been a focus of a lot of research, although most of that research was concerned with either moving objects or data streams, but not with both.

Spatio-temporal databases have been a focus of research in an attempt to define appropriate spatio-temporal data types and algebra that would satisfy all requirements for spatio-temporal data management [1–5]. If these spatio-temporal databases support objects whose position and/or extent change continuously, then we are talking about moving objects databases. In such databases, it is possible to pose queries not just about the history of movements, but also about the current and future movements [6]. Frequent updates of moving objects' positions, fluctuations in the data volume and the requirements for real-time responses

* Corresponding author. Tel.: +385 1 6129 915.
*E-mail addresses:* zdravko.galic@fer.hr (Z. Galić), mirta.baranovic@fer.hr (M. Baranović), kresimir.krizanovic@fer.hr (K. Križanović), emir.meskovic@untz.ba (E. Mešković).

http://dx.doi.org/10.1016/j.datak.2014.02.002
0169-023X/

to posed spatio-temporal queries, have been indicated as serious and perhaps even insurmountable obstacles for the traditional spatio-temporal databases [7].

Several research projects provided prototype systems that manage large and possibly unbounded data streams and continuous queries, as well as common relations and one-time queries. Research efforts in data stream management systems (DSMS) focused mainly on processing continuous queries over data streams, based on a simple relational paradigm. More recently, DSMSs that support data stream management and that can pose continuous queries have been developed [8–10]. These DSMSs support query processors that enable continuously adaptive query processing and handling of large numbers of continuous queries posed on large and varying data streams. However, geospatial and temporal properties of both data streams and continuous queries have been disregarded, and most of the current DSMSs offer very rudimentary support for geospatial data. Some DBMS vendors [11–14] offer data stream processing engines, but only a few of them support geospatial data types and operations by including geospatial toolkit [11], spatial cartridge [15] or extensibility framework to invoke methods from a spatial library [16–18].

For modeling and querying spatio-temporal objects, there exists a comprehensive framework of abstract data types for objects moving freely in 2D space, providing data types such as moving point and moving region [19]. This framework allows one to model movements of objects in a database and to pose queries about the history of such movements.

In a DSMS, data arrives in the form of concurrent and continuous data streams. Queries on these data streams are typically continuous monitoring queries, involving both persistent relations and other time-varying streams, and emitting streaming data in real time as results. Data stream management systems have several significant characteristics different from those of the traditional DBMSs [20]:

(i) Data streams are sequences of records, ordered by the arrival time or by another ordered attribute such as generation time (which is likely to be correlated with, but not equivalent to the arrival time), that arrive for processing over time instead of being available a priori.
(ii) They are produced by a variety of external sources, meaning that a DSMS has no control over the arrival order or the data rate.
(iii) They are produced continually and, therefore, have unbounded, or at least unknown length. Thus, a DSMS may not know if or when the stream "ends".

Our goal is to achieve an integrated query processing approach that runs SQL-like queries continuously and incrementally over data before that geospatial data is stored in the database. The key concept here is that streaming data and persistent data are not intrinsically different — the persistent data is simply streaming data that has been entered into the persistent structures. In other words, queries could be posed exclusively on relations, exclusively on streams, or on a combination of streams and relations [21]. In the first case, a query produces a relation as an output and has the exact same semantics as in SQL. We refer to queries over relations as relational *snapshot* queries because they operate on a snapshot of the relations at a given time — they produce an answer and they terminate. In other cases however, a query produces a stream as an output. Since a stream is unbounded, a query that produces a stream never ends and is therefore called a *continuous* query. To efficiently support geospatial data streams we need a DSMS that supports spatio-temporal data types and operations for geospatial objects.

In general, the main contributions of the paper can be summarized as follows:

(i) We describe a novel formal framework and an implementation approach to the management of spatio-temporal objects considering them as geospatial data streams.
(ii) We are bridging the formal frameworks of geospatial databases, spatio-temporal databases and data streams, at the abstract level.
(iii) We design a type system, at the abstract level, by adopting *multi-sorted* algebra.
(iv) We show the embedding of proposed data types and operations into an SQL-like language.
(v) We describe an implementation approach, based on the UDAF paradigm.
(vi) We prove the proposed formal framework by a developed prototype in a pure DSMS environment.

The remainder of this paper is structured as follows. Section 2 presents an overview of related work in the research areas of spatio-temporal databases, moving objects databases and relational and spatio-temporal data streams. In Section 3, we formally describe a type system and in Section 4, a set of operations on these types. In Section 5, we describe a strategy for the implementation of the spatio-temporal data types and operations, including their embedding into an SQL-like language. Finally, Section 6 concludes the paper and discusses future research directions.

## 2. Related work

In the last two decades, many spatio-temporal data models and corresponding query languages have been studied and proposed in the database research community. Spatio-temporal database perspective and composite spatio-temporal data types were explored in the CHOROCHRONOS project [2], with the goal of managing time-dependent geometries in a database. The research work in that area generally focused on geometries changing in *discrete* steps. This restriction was dropped and *continuously* moving objects (points, lines, and polygons) were considered. This resulted in a data model capturing the complete history of movement with a related query language. The field of the moving objects databases was pioneered by two parallel developments with different perspectives of the field. These can be characterized as the spatio-temporal database perspective and the trajectory location management perspective.