PP-OMDS: An Effective and Efficient Framework for Supporting Privacy-Preserving OLAP-based Monitoring of Data Streams

Alfredo Cuzzocrea¹, Assaf Schuster² and Gianni Vercelli³
¹University of Trieste and ICAR-CNR, Trieste, Italy
²Technion, Haifa, Israel
³University of Genoa, Genoa, Italy

Keywords: Privacy-Preserving OLAP over Data Streams, OLAP-based Monitoring of Data Streams, Privacy-Preserving OLAP-based Monitoring of Data Streams.

Abstract: In this paper, we propose PP-OMDS (Privacy-Preserving OLAP-based Monitoring of Data Streams), an innovative framework for supporting the OLAP-based monitoring of data streams, which is relevant for a plethora of application scenarios (e.g., security, emergency management, and so forth), in a privacy-preserving manner. The paper describes motivations, principles and achievements of the PP-OMDS framework, along with technological advancements and innovations. We also incorporate a detailed comparative analysis with competitive frameworks, along with a trade-off analysis.

1 INTRODUCTION

The PP-OMDS (Privacy-Preserving OLAP-based Monitoring of Data Streams) framework focuses the attention on the research challenge represented by models, techniques and algorithms for supporting privacy-preserving OLAP-based monitoring of data streams. The investigated research context includes and integrates three distinct components, namely privacy-preserving OLAP (e.g., (Peroul et al., 2000; Wang et al., 2004a; Wang et al., 2004b; Agrawal et al., 2005; Hua et al., 2005; S.Y. Sung et al., 2006)), data stream monitoring (e.g., (Bulut et al., 2009; Gan & Dai, 2014; K.-Y. Cao et al., 2014; Huang et al., 2006; Huang et al., 2007)), and privacy-preserving data stream monitoring (e.g., (Dwork et al., 2010b; T.-H.H. Chan et al., 2010; D.J. Mir et al., 2011; Dwork et al., 2006; Beimel et al., 2008; Chen et al., 2012; Friedman et al., 2014)). Even if some sporadic works on supporting aggregate monitoring queries in a privacy-preserving manner exist (e.g., (Shi et al., 2011; Fan & Xiong, 2012)), the three related issues have not been investigated together under the umbrella of a common framework specially focused to OLAP analysis (rather than the underlying aggregate querying layer), which indeed defines a powerful reference application scenario for a wide spectrum of emerging applications over data streams, such as security (e.g., (Zhou et al., 2014; R.H. Deng et al., 2014)) and emergency management (e.g., (Gan & Dai, 2014; Keren et al., 2014)).

Starting from this main motivation, the PP-OMDS framework aims at introducing models, techniques and algorithms for supporting privacy-preserving OLAP-based monitoring of data streams, which is relevant for modern distributed environments such as Clouds. This will fulfill actual limitations of state-of-the-art solutions that do not address the relevant application scenario represented by using OLAP tools and methodologies to support data stream monitoring in a privacy-preserving manner. Application scenarios of the PP-OMDS are many-fold (e.g., security and emergency management). Tangible results of the PP-OMDS framework in real-life applicative settings are represented by relevant research and innovation advancements in the context of models, techniques and algorithms for supporting privacy-preserving OLAP-based monitoring of data streams, which represent the main “result” of the framework. It should be noted that, nowadays, these topics play a critical role as they are “naturally” compliant with the EU H2020 research framework under the topic Big Data.

The methodology implemented within the PP-OMDS framework foresees a multi-step approach that comprises: (i) conceptual analysis and design of case studies and related use cases showing in details how and according to which tasks users/applications
interact with the PP-OMDS framework; (ii) conceptual analysis and design of models, techniques and algorithms for supporting privacy-preserving OLAP-based monitoring of data streams, which provides a top-down overview of the different PP-OMDS framework’s components to be defined and (prototypically) implemented; (iii) design of the OLAP-based models and related algorithms for supporting the monitoring of data streams, which represents a more-detailed view of the activity (ii) specifically focused on OLAP-analysis aspects of the PP-OMDS framework; (iv) design of the privacy-preserving version of the OLAP-based models and related algorithms for supporting the monitoring of data streams, which represents a further refinement of models and algorithms defined by the activity (iii) but targeted to embedding privacy-preserving aspects of the PP-OMDS framework in such models and algorithms; (v) integration of the OLAP component and the privacy-preserving OLAP component, both oriented to data stream monitoring, into the PP-OMDS framework, according to several alternative computing models.

The scientific contribution of the research carried-out by the PP-OMDS framework is very high because state-of-the-art research, even though focused on the relevant problem of privacy-preserving OLAP over static data, lacks of proposals that are specifically focused on privacy-preserving OLAP over data streams, and, in addition to this, there is not a specific integration with monitoring aspects, which, contrary to this, are indeed very relevant for a wide spectrum of next generation data stream applications and systems.

Summarizing, the main research topics investigated by the PP-OMDS framework are the following:

- models, techniques and algorithms for supporting OLAP-based monitoring of data streams;
- models, techniques and algorithms for supporting the privacy-preserving version of OLAP-based monitoring of data streams;
- integration of models, techniques and algorithms devised in the context of the two previous topics/activities, according to several alternative computing models.

The paper describes motivations, principles and achievements of the PP-OMDS framework, along with technological advancements and innovations. We also incorporate a detailed comparative analysis with competitive frameworks, along with a trade-off analysis.

2 RELATED WORK

In this Section, we first review the scientific and technological background of the three distinct PP-OMDS components, namely: privacy-preserving OLAP, data stream monitoring, and privacy-preserving data stream monitoring.

Privacy-preserving OLAP over data streams is a research challenge that puts down roots in the basic privacy-preserving OLAP problem over static (e.g., relational) data (e.g., (Cuzzocrea et al., 2008; Cuzzocrea & Saccà, 2012)). Basically, the latter problem origins from recognizing that malicious users can infer sensitive knowledge from online corporate data cubes that do not adopt effective privacy preserving countermeasures (e.g., (Wang et al., 2004a; Wang et al., 2004b)). From this breaking evidence, which originally derives from the privacy-preserving database management problem (e.g., (Sweeney, 2002; Machanavajjhala et al., 2007)), a plethora of Privacy Preserving Data Mining (PPDM) (Agrawal et al., 2000) techniques has been proposed during the last years. Each of these techniques focuses on supporting the privacy preservation of a specialized KDD/DM task such as frequent item set mining (e.g., (Li et al., 2012)), clustering (e.g., (Alotaibi et al., 2014)), and so forth. Privacy-preserving OLAP (Agrawal et al., 2005) is a specific PPDM technique dealing with the privacy preservation of data cubes (Gray et al., 1997). Data cubes play a leading role in Data Warehousing (DW) and Business Intelligence (BI) systems, as, on the basis of a multidimensional and a multi-resolution vision of data, data cubes make available to OLAP users/applications SQL aggregations (e.g., SUM, COUNT, AVG etc.) computed over very large amounts of data stored in external sources (e.g., relational databases). These aggregations enable OLAP users/applications to easily extract summarized knowledge from the underlying massive data sources, with performance infeasible for traditional OLTP processes (e.g., (Cuzzocrea et al., 2009; Cuzzocrea & Matrangolo, 2004)).

Unfortunately, as highlighted by recent studies (Pernul et al., 2000; Wang et al., 2004a; Wang et al., 2004b; Agrawal et al., 2005; Hua et al., 2005; S.Y. Sung et al., 2006), the privacy risk heavily affects online-published data cubes. By accessing and querying data cubes, malicious users can infer OLAP aggregations computed over sensitive ranges of multidimensional data that, due to privacy reasons, are hidden to unauthorized users. Specifically, since OLAP deals with aggregate data and summarized knowledge, malicious users are usually interested in
inferring what we define as aggregate patterns of multidimensional data, rather than individual information of data cells stored in data cubes (e.g., (Sung et al., 2006)) or tuples stored in relational databases (e.g., (Sweeney, 2002; Muchanavajjhala et al., 2007)). Hence, the final goal of privacy-preserving OLAP techniques consists in protecting the privacy of aggregate patterns of sensitive interest for the target organization.

Directly deriving from the static case, the dynamic case, i.e. OLAP over data streams (e.g., (Cuzzocrea, 2013; Cuzzocrea, 2011; Cuzzocrea, 2009)), extends the basic problem to the more challenging issue of dealing with data streams (e.g., (Babcock et al., 2006)). This introduces harder research challenges to be considered, due to the fact that additional requirements of making privacy-preserving OLAP aggregations computed over data streams occur (e.g., (Cuzzocrea, 2009)), mainly coming from the well-known constraints dictated by data stream processing, i.e. bounded memory, single-pass algorithms, near-real-time computation (e.g., (Golab & M.T. Ozsu, 2003)). Looking at the actual literature, it follows that, while there is considerable amount of research work on the problem of effectively and efficiently executing Data Mining tasks over data streams such as clustering and classification (e.g., (Gaber et al., 2005; C.C. Aggarwal & P.S. Yu, 2008; Zhang et al., 2008)), there exist in literature few research initiatives that focus the attention on the yet-relevant issue of effectively and efficiently supporting OLAP over data streams (e.g., (Cuzzocrea, 2009)), also taking into consideration uncertainty and imprecision that very often characterized such data type (e.g., (Cuzzocrea, 2013; Cuzzocrea, 2011)), which, similarly (as to confirm their relevance), have also been studied in well-known Data Mining settings (e.g., (Cuzzocrea et al., 2014)). (Cormode & Garofalakis, 2007) and (Jayram et al., 2007) are studies that put theoretical foundations to the OLAP over data streams problem, since they focus on the problem of computing aggregates over probabilistic data streams. Indeed, probabilistic data stream models, which have been firstly introduced in (L.J.C. Re et al., 2010), are well suited to represent inherent characteristics of data streams. Unfortunately, these approaches do not focus on specific features of streaming data, such as multidimensionality (Cuzzocrea, 2009), which is indeed critical in data stream analysis (Cai, 2004; Han et al., 2005). In line with this research initiative, (Cormode & Garofalakis, 2007) and (Jayram et al., 2007) offer theoretical models for understanding and capturing aggregations in probabilistic data streams, mainly via the idea of exploiting Probability Distribution Function (PDF) to model uncertain and imprecise data streams. (M.-J. Hsieh et al., 2007) has some relation with the research related to OLAP over data streams, due to the fact it studies the issue of compressing stream cubes by inheriting well-known models and algorithms for approximate query processing, with the idea of adding more efficiency and reliability. Lahar (L.J.C. Re et al., 2010) is a specialized warehousing system that focuses on so-called Markovian streams, which well model and capture uncertainty and imprecision in streaming data, and proposes approximate methods for answering event-OLAP queries over such class of data streams via trading-off efficiency and accuracy. (Jampani et al., 2008) is also relevant to our research due to the fact it introduces a nice Monte-Carlo-based approach for managing uncertain data that could be easily incorporated in our proposed PP-OMDS framework in order to find finer probabilistic models.

Data stream monitoring is an active area of research where the main issue consists in devising models, techniques and algorithms for supporting the online monitoring of streaming data, which is relevant for a wide spectrum of applications ranging from telecommunication networks to web-click streams, from intrusion detection systems to sensor networks, and so forth. As highlighted by recent studies (e.g., (Bulut, et al., 2009)), the main research challenge to deal-with in this context is represented by the strong need for efficiency in terms of space usage and per-item processing time while providing high-accuracy answers to several classes of useful queries, such as aggregate monitoring queries, aimed at finding surprising levels in the target data stream, and similarity queries, devoted to detect correlations and find interesting patterns in the target data stream. This research line has originated a rich collection of proposals that focus the attention on the specific data stream monitoring problem. (Gan & Dai, 2014) proposes a technique for detecting dynamic changes in frequencies of episodes over time-evolving streams, with special emphasis on online detection of emerging patterns such as abrupt emerging episodes and abrupt submerging episodes (over streams). (K.-Y. Cao, et al., 2014) proposes Continuous Uncertain Outlier Detection (CUOD), a framework capable of quickly determining the nature of uncertain elements in data streams, by exploiting pruning strategies in order to improve the efficiency. Furthermore, a further alternative pruning approach, named as Probability Pruning for Continuous Uncertain Outlier Detection (PCUOD), is proposed to reduce the detection cost. The latter is a method for
probabilistically estimating outliers in streams, in order to effectively reduce the amount of computational overhead. Communication-efficient monitoring of distributed streams has also been the subject of much research in recent years. Some research has focused on anomaly detection (Huang et al., 2006; Huang et al., 2007), while other studies focus on monitoring specific types of functions, including sums (Keralapura et al., 2006; Olston et al., 2003), Boolean predicates (Agrawal et al., 2006), inner products (Cormode & M.N. Garofalakis, 2008) and entropy (Arackaparambil et al., 2008). (Sharfman et al., 2006) employs techniques presented in the context of geometric monitoring, which enable monitoring arbitrary threshold functions by interpreting the monitoring task as a geometric problem. This research has been important extensions recently (e.g., (Lazerson et al., 2015; Keren et al., 2014; Giatrakos et al., 2014; Keren et al., 2012)).

Privacy-preserving data stream monitoring is directly related to our research (Keren et al., 2014; Friedman et al., 2014). The application of differential privacy (Dwork, 2006) to data stream processing has been studied initially in (Dwork et al., 2010), which has introduced the concept of pan-private data stream algorithms – algorithms that retain their privacy properties even when intrusions ex-pose the internal state of the system. Two independent works (Dwork et al., 2010b) and (T.-H.H. Chan et al., 2010) study continuous release of differentially-private counts, optionally while ensuring pan-privacy. Mir et al. (Mir et al., 2011) rely on sketches to track statistics such as distinct count, cropped first moment, and heavy hitters count over fully dynamic data while preserving pan-privacy. While we do not aim to obtain pan-privacy, the PP-OMDS framework could be extended to support it through straightforward application of the technique of (Dwork et al., 2010b). Indeed, exploiting differential-privacy tools (e.g., (Dwork, 2006a)), combined with OLAP over data streams (e.g., (Cuzzocrea, 2009)), with monitoring goals (e.g., (Keren et al., 2014; Friedman et al., 2014)) is very promising for achieving the PP-OMDS project’s goals. Early works on differential privacy in a distributed setting (Dwork et al., 2006, Beimel et al., 2008) study how differential privacy could be combined with cryptographic protocols to allow one-off computations to be carried out both securely and privately. Chen et al. (Chen et al., 2012) make use of the noise generation mechanism of (Dwork et al., 2006) to allow analysts to pose histogram queries to a subset of distributed clients with the help of an honest but curious proxy. (Friedman et al., 2014) proposes a general framework that enables monitoring arbitrary functions over statistics derived from distributed data stream in a privacy-preserving manner.

Several works study differentially-private aggregation over distributed time-series data, focusing mostly on simple aggregates, such as counts and sums. Rastogi and Nath (Rastogi et al., 2010) rely on the Discrete Fourier Transform to compress historical time-series data, and leverage threshold homomorphic encryption to run a distributed version of the Laplace mechanism on the compressed data. As the compression requires access to all the query results in advance, this method is not adequate for processing data streams on the fly. Shi et al. (Shi et al., 2011) apply cryptographic techniques to allow an untrusted aggregator to compute differentially-private sums over distributed peers without learning anything but the outcome. While the proposed scheme has been designed to reduce the overhead of cryptographic operations in periodical communications, it does not address the cumulative privacy loss. Fan and Xiong (Fan & Xiong, 2012) address the dynamic nature of the data by adaptive sampling of the time-series data and use of Kalman filters for estimating the data in non-sampling points.

3 REFERENCES APPLICATION SCENARIO

In the PP-OMDS framework, the main research focus is on the issue of supporting privacy-preserving OLAP-based monitoring of data streams, which, has highlighted above, is innovative in actual state-of-the-art research, and it is of relevant interest for a wide spectrum of data stream applications (e.g., security (Zhou et al., 2014; R.H. Deng et al., 2014), emergency management (Gan & Dai, 2014; Keren et al., 2014), and so forth). Figure 1 shows a typical application scenario for the PP-OMDS framework. Here, a sensor network, composed by both sink nodes and sensor nodes (e.g., (Cuzzocrea, 2014)), produces sensor readings of kind (ID, t, v) such that: (i) ID is the absolute identifier of the sensor node, (ii) t is the timestamp at which the sensor reading is produced, (iii) v is the proper reading (i.e., the value). A privacy-preserving 2D OLAP-based monitoring view V is interfaced to the sensor network directly, and it is used to monitor the reading value variable v, based
on a complex multidimensional data model (Gray et al., 1997; Cuzzocrea, 2009). In the reference application scenario of Figure 1, the model introduces Location as the first dimension and Time as the second dimension, respectively, and the measure value is defined on top of a COUNT aggregate operator over a two-dimensional range \((r_1, r_2)\), being \(r_1\) and \(r_2\) two one-dimensional ranges along Location and Time, respectively, which define the data cube cell. The variable is monitored via suitable aggregate monitoring queries (e.g., (Bulut, et al., 2009)) over data cube cells that populate the OLAP view \(V\).

With reference to the so-delimited application scenario, a critical challenge relies on effectively and efficiently computing the OLAP view \(V\) over sensor readings (i.e., streaming data), with the additional requirement that \(V\) must be computed in a privacy-preserving manner (e.g., (Cuzzocrea et al., 2008; Cuzzocrea & Saccà, 2012) – for the static case), i.e. it must preserve the privacy of data sources that originate the sensor readings. The OLAP view \(V\) is meant for monitoring goals: when the COUNT-aggregated value of a data cube cell \(C_i\) exceeds a fixed threshold \(t\), then an event occurs and a suitable action (trigger, respectively) is activated in order to modify the target monitored environment. Obviously, the described application scenario is just an instance that represents even more sophisticated multidimensional settings, where the privacy-preserving OLAP-based monitoring view is characterized by multiple dimensions (e.g., (Han et al., 2005)). It is worth to recognize that the described application scenario well-describes a plethora of modern data stream applications, ranging from event detection (e.g., (Gyu Kim & Kim, 2014)) to complex monitoring queries support (e.g., (Li, 2014)), from near-duplicate detection over multimedia streams (e.g., (C.-Yi Chiu et al., 2013)) to anomaly detection (e.g., (Taylor et al., 2013)), and so forth, all with the innovative and challenging requirement of preserving the privacy of data streams (e.g., (Al-Hussaeni et al., 2014; Kim et al., 2014)), under OLAP analysis requirements (e.g., (Cuzzocrea, 2013; Cuzzocrea, 2011)), with monitoring purposes (e.g., (Keren et al., 2014; Friedman et al., 2014)), being the latter the three main pillar of the PP-OMDS project.

4 IMPACT AND EXPLOITATIONS

The PP-OMDS framework provides relevant research and innovation advancements in the context of models, techniques and algorithms for supporting privacy-preserving OLAP-based monitoring of data streams, which represent the main “result” carried-out by the framework.

As highlighted in Section 1 and Section 2, where research topics of the PP-OMDS framework, along with the background state-of-the-art work, have been described in details, this will represent a critical advancement of actual literature, because while several proposals on privacy-preserving data stream monitoring exist (e.g., (Dwork et al., 2010b; T.-H.H. Chan et al., 2010; D.J. Mir et al., 2011; Dwork et al., 2006; Beinel et al., 2008; Chen et al., 2012; Friedman et al., 2014)), as well as several proposals on privacy-preserving OLAP, the majority on static data (e.g., (Cuzzocrea et al., 2008, Cuzzocrea & Saccà, 2012)), and few proposal of OLAP over streaming data (e.g., (Cuzzocrea, 2013; Cuzzocrea, 2011; Cuzzocrea, 2009)), research efforts have not investigated, till now, the yet relevant challenge of combining the basic components under the vision of a common framework specially focused to privacy-preserving OLAP-based monitoring of data streams.
Along with the models, techniques and algorithms above, the PP-OMDS framework also outcomes other contributions, such as:

- theoretical complexity analysis (asymptotic issues and possible lower bounds);
- properties and lemma deriving from the main theory;
- computational paradigms;
- reference case studies and application scenarios;
- performance analysis of devised algorithms;
- robustness and sensitivity analysis of devised algorithms;
- optimization strategies and solutions for devised algorithms.

Given the specificities of the investigated problem, even theoretical privacy-analysis-oriented contributions are expected. These comprise:

- theoretical analysis of the privacy-preserving capabilities of devised models and techniques;
- analysis and detection of possible privacy breaches (e.g., (Chi-Wing Wong et al., 2011));
- analysis of the robustness of devised models and techniques against harder attack configurations such as coalitions (e.g., (Cuzzocrea & Bertino, 2011)).

All these contributions clearly represent a milestone for future research efforts in the field, which can be reasonable intended as one of the fundamental expected impacts of the PP-OMDS framework. Moreover, as already highlighted, the PP-OMDS framework is “naturally” compliant with the topics carried-on within the EU H2020 research framework, as they essentially investigate the topic Big Data, which is very relevant within H2020 with a relevant number of calls focused on it in all the main (H2020) axes, i.e. Excellence Science, Industrial Leadership, Societal Challenge.

In addition to the main research-oriented impact, the PP-OMDS framework also provides a number of technological expected outcomes, which can be summarized as reference technical and algorithmic solutions for effectively and efficiently supporting privacy-preserving OLAP-based monitoring of data streams in critical data-intensive settings, such as:

- distributed processing of streaming financial data in inter-organizational settings (e.g., like the ones built upon emerging Cloud Computing infrastructures) – in this application scenario, participants (e.g., bank, insurance, government agency, and so forth)

would like to disclose aggregate information for inter-organizational processing support (e.g., monitoring pre-defined balance thresholds) while, at the same, still protecting aggregates over their own sensitive ranges of data;

- distributed processing of streaming social inter-network data for analysis purposes – in this application scenario, streaming data coming from social inter-networks are monitored and analyzed according to OLAP-based multidimensional models in order to discover useful knowledge (e.g., anomalies, alarms, emerging patterns, user profiling and user tracking metrics, and so forth) focused to the interaction of different social networks, even defined on top of heterogeneous semantic domains.

5 PP-OMDS ARCHITECTURE AND MAIN FUNCTIONALITIES

Figure 2 shows the detailed architecture of the PP-OMDS framework, along with its components and reference technologies. These are:

- **Data Stream Source Layer** – The layer where data streams are produced by a target process (e.g., sensor networks, logistic networks, etc.). Technology used: arbitrary (at the input layer).

- **Data Stream Acquisition Server** – The server devoted to acquire data streams (by also providing the initial pre-processing – e.g., cleaning) and to make the necessary data model transformations. Technology used: Java programming language; JBoss application server, MySQL database server. In particular: (i) JBoss provides the necessary Java execution runtime environment for the Data Stream Collection Component, which is the component that implements classes and functions supporting the data stream acquisition phase; (ii) My SQL database server provides the necessary data storage (even buffer-oriented) and management functionalities.

- **Data Stream Monitoring Server** – The server devoted to the monitoring of data streams, by implementing the algorithms proposed in the PP-OMDDS framework. It fully interacts with the Data Stream OLAP Server and the
Data Stream Privacy-Preservation Server, respectively, in order to achieve the overall privacy-preserving OLAP-based monitoring of data streams pursued by the PP-OMDDS project. Technology used: Java programming language; JBoss application server. In particular, JBoss provides the necessary Java execution runtime environment for the OLAPing Data Stream Component, which is the component that implements classes and functions supporting OLAP analysis over data streams; (ii) Mondrian OLAP server provides the necessary multidimensional data storage and management functionalities.

Data Stream Privacy-Preservation Server – The server devoted to support privacy-preserving management of data streams, by implementing the algorithms proposed in the PP-OMDDS project. It fully interacts with the Data Stream Monitoring Server and the Data Stream OLAP Server, respectively, in order to achieve the overall privacy-preserving OLAP-based monitoring of data streams pursued by the PP-OMDDS project. Technology used: Java programming language; JBoss application server. In particular: (i) JBoss provides the necessary Java execution runtime environment for the Privacy-Preserving Data Stream Management Component, which is the component that implements classes and functions supporting the privacy-preserving data stream management phase.

Analytics Server – The server devoted to support data stream analytics, according to the privacy-preserving OLAP-based vision pursued by the PP-OMDDS framework. Technology used: Java programming language; JBoss application server; MySQL database server. In particular: (i) JBoss provides the necessary Java execution runtime environment for the PP-OMDS Analytics Component, which is the component that implements classes and functions supporting the PP-OMDS analytics phase; (ii) MySQL database server provides the necessary data storage (even buffer-oriented) and management functionalities.

Streaming Analytics Layer – The layer where streaming analytics functionalities are finally implemented and delivered to the client applications-users, based on specific analytics goals (e.g., emergency detection and management, security, etc.). Technology used: arbitrary (at the output layer).

Figure 2: PP-OMDS Framework Detailed Architecture.
The PP-OMDS framework exploits software and hardware solutions that are currently available and completely technologically-feasible (even considering open-source solutions, in the case of software). Indeed, looking at the hardware architecture of the PP-OMDS framework, we identify an (hardware) architecture that makes use of standard solutions for (i) collecting data streams, (ii) monitoring data streams, (iii) aggregating data streams, (iv) providing privacy-preserving methods over data streams, and (v) supporting OLAP-like query answering over data streams. All these components are commonly delivered on top of well-known architectures composed by (i) data servers, (ii) OLAP servers, (iii) application servers, (iv) client components (e.g., desktop computers, laptops, mobile devices, etc.). As regards the software architecture of the PP-OMDS framework, we identify a (software) architecture populated by software components that can be developed by means of standard high-level programming languages (e.g., Java, C++, etc.) and standard data access and manipulation methods (e.g., JDBC, ODBC, etc.). It should be noted that, for both the hardware architecture and the software architecture of the PP-OMDS framework, the components to be developed adhere to well-assessed and mature technologies (both hardware and software) that clearly make the PP-OMDS framework completely-feasible. In addition to this, all the components of the PP-OMDS framework are already Cloud-enabled, hence the framework can be easily extended towards a Cloud-based application (hence, improving performance, reliability and availability).

Table 1: Comparative Analysis of Competitor Frameworks and the PP-OMDS Framework.

<table>
<thead>
<tr>
<th></th>
<th>Distributed Data Stream Monitoring</th>
<th>OLAP Analysis over Data Streams</th>
<th>Privacy-Preserving Data Stream Management</th>
<th>Support for General-Purpose Applications</th>
<th>Support for Vertical Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>APAMA Streaming Analytics</td>
<td>✓</td>
<td>x</td>
<td>x</td>
<td>✓</td>
<td>x</td>
</tr>
<tr>
<td>Event Stream Processor</td>
<td>✓</td>
<td>x</td>
<td>✓</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Apache Spark</td>
<td></td>
<td></td>
<td>✓</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>PP-OMDS</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

6 COMPARATIVE ANALYSIS WITH COMPETITIVE FRAMEWORKS

The software product context of the PP-OMDS framework is represented by the wide area of streaming analytics tools and systems, and, in particular, those devoted to support stream monitoring. Nevertheless, it does not exist a solution that specifically focuses on the relevant problem of supporting privacy-preserving OLAP-based monitoring of distributed data streams, as delineated by the PP-OMDS framework’s motivations. This confirms to us the innovativeness of our proposal.

In the following, we focus the attention on the specific block-founding problems that characterizes the PP-OMDS framework. For what regards the basic distributed data stream monitoring problem, some relevant tools and systems that are currently available are the following ones:

- **APAMA Streaming Analytics**, from Software AG;
- **Stream Computing**, from IBM;
- **Event Stream Processor**, from SAP;
- **Apache Spark**, from Cloudera.

For what regards both the basic OLAP analysis over data streams problem and the basic privacy-preserving data stream management problem, there do not exist direct tools and systems currently available, although vertical solutions on top of existing streaming analytics platforms can be devised. The most critical limitation of competitor frameworks is represented by the fact that they all adhere to a common deployment model, i.e. providing a general platform for supporting data stream analytics. While on top of such (common)
platform personalized solutions can still be developed (via ad-hoc IDE and functional programming environments), they do not focus on the specialized problem of supporting privacy-preserving OLAP-based monitoring of distributed data streams, despite the relevance of this problem and its target application scenarios (e.g., emergency management, security, and so forth).

On the other hand, developing personalized solutions on general-purpose streaming analytics platforms poses critical challenges for what regards a wide spectrum of issues, ranging from complexity overheads to (software) maintenance problems, from heterogeneous data format integration issues to analytical front-end tools that are not focused to the peculiarities of the target application (thus making the whole knowledge discovery from big streaming data harder), and so forth. All these considerations clearly suggest the adoption of an all-inside, self-contained analytical framework that, still being focused on the specific goal of supporting privacy-preserving OLAP-based distributed data stream monitoring, can be further specialized on particular, vertical application settings (still falling in the reference technological area).

In addition to this, the competitor frameworks do not provide an explicit support neither to OLAP analysis tools over data streams neither to privacy-preserving data stream management, making for them hard to follow the paradigms dictated by the PP-OMDS framework’s paradigms. Table 1 proposes a summary on the comparative analysis of the competitor frameworks against the PP-OMDS framework, along the above-discussed parameters/functionalities.

7 CONCLUSIONS

In this paper, we propose have proposed PP-OMDS, an innovative framework for supporting the OLAP-based monitoring of data streams, which is relevant for a plethora of application scenarios (e.g., security, emergency management, and so forth), in a privacy-preserving manner.

REFERENCES

A. Cuzzocrea, U. Matrangolo: Analytical Synopses for Approximate Query Answering in OLAP Environments, DEXA, 2004
A. Cuzzocrea: Approximate OLAP Query Processing over Uncertain and Imprecise Multidimensional Data Streams. DEXA, pp. 156-173, 2013
A. Cuzzocrea: Optimization issues of querying and evolving sensor and stream databases, Inf. Syst. 39:196-198, 2014
A. Cuzzocrea: Retrieving Accurate Estimates to OLAP Queries over Uncertain and Imprecise Multidimensional Data Streams. SSDBM, 2011, pp. 575-576
A. Friedman, I. Sharfman, D. Keren, A. Schuster: Privacy-Preserving Distributed Stream Monitoring, NDSS, 2014
C. Arackaparambil, J. Brody, A. Chakrabarti: Functional monitoring without monotonicity. ICALP, 2009
C. Dwork, M. Naor, T. Pitasii, G.N. Rothblum, S. Yekhanin: Pan-private streaming algorithms. ICS, 2010
C. Dwork, T. Pitassi, M. Naor, G.N. Rothblum: Differential privacy under continual observation, STOC, 2010
C. Dwork: Differential Privacy, ICALP (2) 2006, pp. 1-12


G. Li, C. Luo, J. Li: Continuous Monitoring of Top-k Dominating Queries over Uncertain Data Streams. *WISE*, 2014, pp. 244-255


I. Sharfman, A. Schuster, D. Keren: A geometric approach to monitoring threshold functions over distributed data streams, *SIGMOD*, 2006


J. Han, Y. Chen, G. Dong, J. Pei, B.W. Wah, J. Wang, Y.D. Cai: Stream Cube: An Architecture for Multi-Dimensional Analysis of Data Streams, *Distributed and Parallel Databases* 18(2), 2005


M. Gaber, A. Zaslavsky, S. Krishnaswamy: Mining Data Streams: A Review, *SIGMOD Record* 34(2), 2005


M. Hua, et al. FMC: An Approach for Privacy Preserving OLAP. *DaWaK*, pp. 408-417, 2005

M.-J. Hsieh, M.-S. Chen, P.S. Yu: Approximate Query Processing in Cube Streams, *IEEE Trans. on Knowledge and Data Engineering* 19(11), 2007


R. Chi-Wing Wong, A. Wai-Chee Fu, K. Wang, P.S. Yu, Jian Pei: Can the Utility of Anonymized Data be Used for Privacy Breaches? *ACM Trans. on Knowledge Discovery from Data* 5(3):16, 2011


V. Rastogi, S. Nath: Differentially private aggregation of distributed time-series with transformation and encryption, *SIGMOD*, 2010

Y.D. Cai, D. Clutterx, G. Papex, J. Han, M. Welgex, L. Auvilx: MAIDS: Mining Alarming Incidents from Data Streams, *ACM SIGMOD*, 2004