A Model of a System for Stream Data Storage and Analysis Dedicated to Sensor Networks of Embankment Monitoring

Anna Pięta, Michał Lupa, Monika Chuchro, Adam Piórkowski, and Andrzej Leśniak

Department of Geoinformatics and Applied Computer Science
AGH University of Science and Technology
al. Mickiewicza 30, 30-059 Cracow, Poland
{chuchro,apieta}@geol.agh.edu.pl,
{mlupa,pioro,lesniak}@agh.edu.pl
http://www.geoinf.agh.edu.pl

Abstract. Contemporary monitoring systems are a source of data streams. Processing of this data is an interesting issue from both a performance and data storage perspective. It is worth paying attention to the concept of stream database management systems, which are a hybrid that allows for efficient analysis of the data stream and provide a set of implemented statistical methods.

This article presents the issues raised by embankment monitoring systems, a sensor network which generates a large stream of measured signals that should be analyzed in real-time. Warning or crash scenarios are also generated which are compared to the incoming data. A model of this data is presented and a construction of stream data analysis is proposed.

Keywords: stream databases, sensor networks, time series.

1 Introduction

The protection of built-up areas against flooding poses modern technology with a very important task. Despite progress, every year floods occur both in Poland and around the world. Some phenomena are sudden and unexpected, while others are avoidable through the construction and control of embankments. Monitoring systems of environmental hazards have been the subject of research, both in terms of flooding [1,15,22,26] and landslides [7,8].

There are many types of monitoring systems. While there are existing solutions, in most cases a dedicated system which incorporates a complex software solution is required.

The monitoring system is usually a source of data streams that represent measurements taken from a line or a grid of points, or, sometimes, from a mobile sink [23]. The amount of data involved is usually large, therefore in the past decade many data stream processing proposals have been put forward, some

K. Saeed and V. Snášel (Eds.): CISIM 2014, LNCS 8838, pp. 514–525, 2014.

of them involving a data stream management system (DSMS). For example, DSMS's such as Aurora [4], Tribeca [25] or GigaScope [6] were created especially for the purposes of telecommunications and networks.

An interesting trend is the subgroup of solutions focused mainly on sensor networks [16,14], including for example, Fjord [17], Cougar [28] or TinyDB [18].

These solutions, although they are no longer under development, are very interesting because they can be used in the implementation of embankment monitoring systems. It is also worth drawing attention to some constantly developed commercial products, such as StreamBase [24]. There are also other, academic, non-commercial projects, such as [11].

1.1 Sensor Network for Embankment Monitoring System

An embankment monitoring system consists of a network of sensors in wells which measure selected physical and geotechnical (e.g. temperature, pressure, pore water, humidity, stress and strain, electrical conductivity, etc.) parameters at given time intervals. The network should cover the embankments on both sides of the riverbed, initially assuming a distance of approximately 5 meters between successive measurement points (Fig. 1). The entire system thus creates a grid of sensors, from which a constant feed of the parameters studied at each measurement point is flowing. This data must then be monitored by software that raises an alarm only if the changes are not caused by the normal diurnal cycle, or seasonal and annual fluctuations.

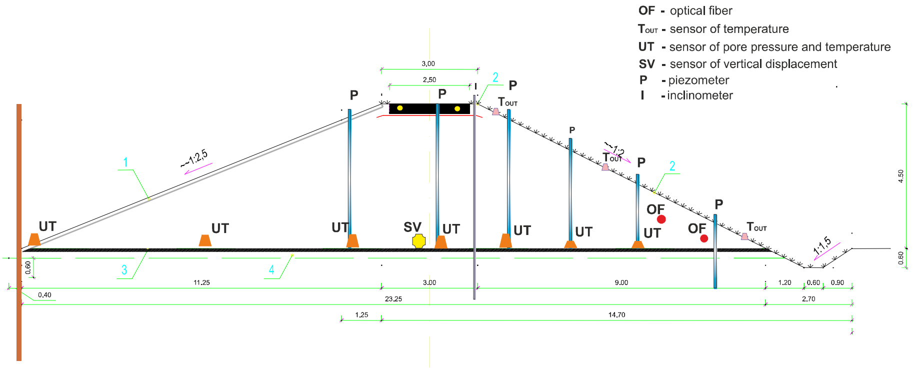

Fig. 1. The schema of experimental flood embankments

2 The Data Model and Query Language for the Dedicated Stream of Data from Sensor Networks

Analysis of the data measured and transmitted over a sensor network placed at given points in embankments is essential in order to assess the condition of the levees and provide early warning about changes in the value of the measured physical parameters. Two different types of data management systems are used in order to implement an embankment's flood control system; a stream database and a relational database. The streaming database is responsible for storing records received from point or linear sensors placed in flood embankments. The data transmitted by the sensors will carry information about parameters such as temperature, saturation and pore pressure that can be collected every minute (Fig. 2a). The relational database collects such parameters as temperature, saturation and pore pressure, obtained by numerical modeling (Fig. 2b). The modeling will be carried according to dynamically determined scenarios created for different initial and boundary conditions, and various mechanical properties of the embankment.

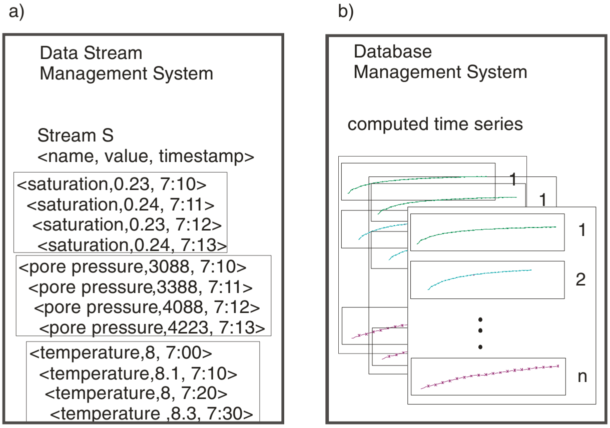

Fig. 2. The two data types used in embankment state monitoring

2.1 Description of Data Streams

The data stream is defined as a set of pairs $< s, t >$, where s is a tuple, and t is a time stamp [3]. Data streams differ from data stored in relational databases because stream data arrives in real time, so the database management system has influence neither on the content nor the order of the data. Another property of the stream data is that there is no restriction on the amount of data. In practice it is assumed that the stream contains an infinite amount of data. Another disadvantage of stream data is that analysis can only be easily carried out after the data has been downloaded from the stream. After data processing decisions concerning data archiving or destruction are taken and the system can proceed to analysis of the next packet from the data stream. A typical query implemented by the stream database can be in the form of an unlimited iteration which is characterized by a sequence of query-response, response, response. The main problem in the processing of stream data is the need to make decisions during the analysis of data packets that continuously arrive at the system database. Operations performed on data recorded by sensors should include a range of operations related not only to their analysis, but also to their initial segregation and processing. The most important operations that should be implemented are:

- selection,
- operations based on nested aggregation that compares the current value retrieved from the data stream with the moving average calculated for the specified range of data,
- operations similar to traditional query-based operations, using union operators and group by clauses, used to merge or decompose data stream
- analysis of the stream corresponding to data mining procedures such as pattern matching, similarity searching, and forecasting,
- operations which combine streams coming from different sensors and display them together with types of static data, queries executed in given time windows [9,10].

2.2 Description of Relational Data

The aim of the designed system is the comparison of data packets obtained from the flood embankments monitoring network with theoretical results obtained from numerical modeling. The calculations were performed using FLAC 2D software v. 7.0 and FLAC 3D v. 5.0 Itasca Consulting Group [12]. Calculations were performed for two-dimensional and three-dimensional models. Numerical calculations were done in order to determine the behavior of the embankments with different initial and boundary conditions. Dynamic module analysis was used to examine the impact of water filtration and temperature changes on the stability of flood embankments. A timed series of parameters such as pore pressure, temperature and saturation were recorded for given points of a computational grid. The governing equation describing the interaction of the water filtration and mechanical processes is described by generalized Biot's theory [2]. The numerical analysis of thermal processes, described by equation (1) incorporates both the conduction and advection processes. The equation describes the phenomenon of the spreading of temperature within the model and the impact of this process on the stresses and strains fields, as well as the advection phenomenon.

$$-\nabla \cdot q^T + q_v^T = \rho C_v \frac{\partial T}{\partial t} \tag{1}$$

where:

- T - temperature,
- q^T - thermal flux,
- q_v^T - volumetric heat source intensity,
- ρ - reference density,
- C_v - specific heat of the fluid.

The numerical simulations of embankment behavior included scenarios that do not affect the stability of embankments, as well as scenarios that led to the violation of their stability. Embankment behavior scenarios were constructed for different sets of initial and boundary parameters and for different physical parameters that reflected the weakening of the embankments. The results of modeling that simulated sensor readings and powered a relational database were used as comparative material for the analysis of the data stream coming from the sensors placed in the real embankment. The modeling process is presented in the figure (Fig. 3).

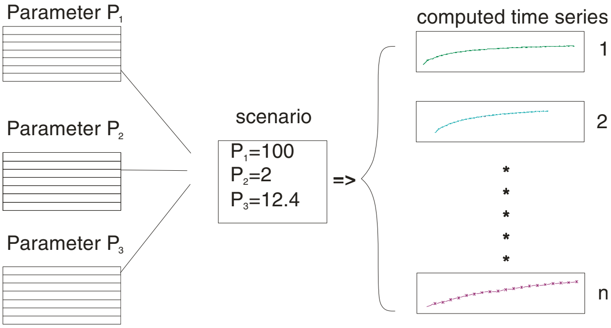

Fig. 3. Generation method of the relational database data used in the analysis step of streaming data. Modeling was conducted for given parameters values that describe the conditions that the embankment is subjected to.

3 Implementation of the System

Analysis of how existing flood embankments behave during floods leads to the conclusion that the state of the fortifications in different parts of the river change dynamically. This is caused by a number of factors, from the size of the flood and the time of its impact on the embankment to the geology and topography of the area. Therefore, one of the main principles of the system is the division of the embankment monitoring running along the river into segments. This procedure will maximize the effectiveness and accuracy of the expected embankment behavior that will be generated by the decision algorithms.

3.1 Stream Data Processing

The measurement network, depending on the section of embankments, will consist of 500-1000 sensors placed at different levels, which will collect data at a specified interval. Therefore, the minimum number of handled streams $S < s, t >$ must be consistent with the number of sensors (˜1000), and the process should not have any delays which lead to a decline in data processing productivity.

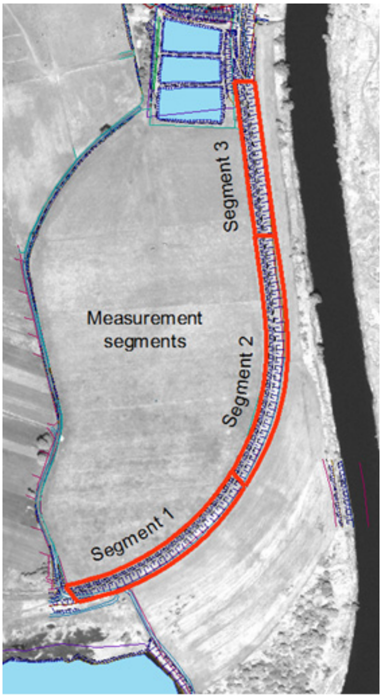

Fig. 4. Arrangement of the measurement segments on the embankment, Czernichow, Poland

A project map of the sensors' locations in the embankment (one segment contains one measurement network) is shown in Figure 4. Data reading will occur through an interface that handles the binary streams that will be generated by the sensor network. It will follow SOA (Service Oriented Architecture) principle where measured data will be processed by services deployed on the sensor nodes and on the server side of the system. Data between services will be transmitted using MoM (Message Oriented Middleware) [27] over IP and GPRS communication protocols [26]. Such a sensor network should adapt itself to the changing execution environment caused by the varying weather conditions that may result, among others, in network connection interruptions and delays [29]. One portion (tuple) s of data will include, in addition to the time stamp t, nine attributes, which will provide the current sensor readings; pore pressure, temperature, stress, etc. An exemplary system scheme is shown in the figure below (Figure 5).

3.2 Processing Data Operators

Each data stream is processed immediately by the implemented operators (logical operators, grouping clauses) that manipulate incoming data, sorting and

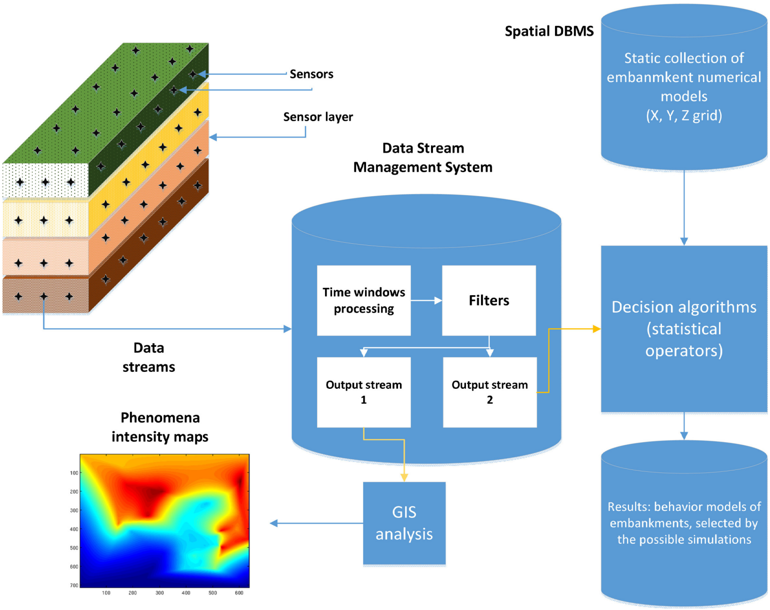
Spatial DBMS
Static collection of embanmkent numerical models (X, Y, Z grid)
Sensors
Sensor layer
Data Stream Management System
Time windows processing
Filters
Output stream 1
Output stream 2
Decision algorithms (statistical operators)
Data streams
Phenomena intensity maps
100
200
300
400
500
600
700
100
200
300
400
500
600
GIS analysis
Results: behavior models of embankments, selected by the possible simulations

Fig. 5. System schema

grouping them according to the size of the measured parameters and the location of the sensor in the embankment. Since the measured values are often noisy due to the conditions prevailing inside the embankments, an important step in processing is a noise reduction process. For this purpose median filtering operators will be implemented, allowing the measurement to be authenticated. The flow of the data stream being processed using the implemented operators is presented in Figure 6. The processed streams are aggregated using time windows that generate an assumed time series, allowing the isolation of the most probable scenarios of embankment behavior by matching the size measurements to static stored (in a relational database) models. Implemented decision algorithms enable the selection of the n-best fit scenarios for the behavior of the embankment, on the basis of matching the measurement time series and modeled data [5].

3.3 DBMS

The decision algorithms proposed in the previous section are on one hand based on the data stream, on the other hand on the behavior of the embankment scenarios, which were prepared based on the numerical modeling. Due to the very high computational complexity, the numerical modeling of the embankment stability is not possible in real time. Therefore, the scenarios are prepared in advance based on many hours of simulations of embankment behavior, using the software package FLAC. Metodology of generation of a single scenario and exampled results are described in [21]. The result of each simulation is a

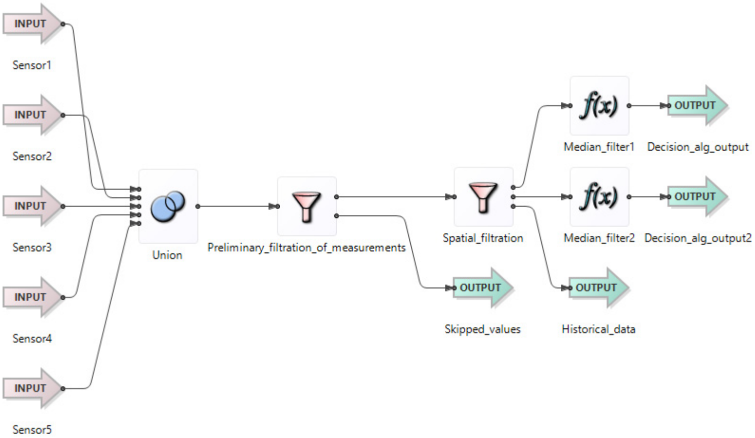

Fig. 6. Stream data processing

scenario and corresponding three-dimensional coordinates grid of the embankment model. Such prepared data sets will supply a relational database, which is the reference point for incoming measurements. The system should also be able to store historical data, ie, the knowledge base, supplied by the data collected during each previous flood. Due to this the decision algorithms will ”learn”, in order to increase their efficiency.

3.4 Used Technologies

Since one of the principles of the system is its reliability, a decision was made to use commercial solutions. The stream database system is StreamBase LiveView (Server, Desktop), which provides an API, enabling customers to implement tools (operators of median filters, time windows, TCP/IP interfaces). A relational database will be created based on MS SQL Server 2014. Numerical modeling is carried out based on the FLAC software.

4 Construction of the Stream Data Analysis System

The data interpretation procedure relies on the synthetic scenarios realized by numerical modeling in a Flac system. The scheme of the procedure is shown in Figure 7. The particular scenario S_{ij} consists of a set of time series. Each of them describes dynamic changes of the measured physical quantity in the particular node of the computational grid. The static, relational data basis stores time series generated for each node of the grid in the assumed time observation window (e.g. two weeks). The data covers a large number (over 500) scenarios constructed for different initial and edge conditions. Let's assume that the measurements are taken using N digital sensors mounted in the embankment.

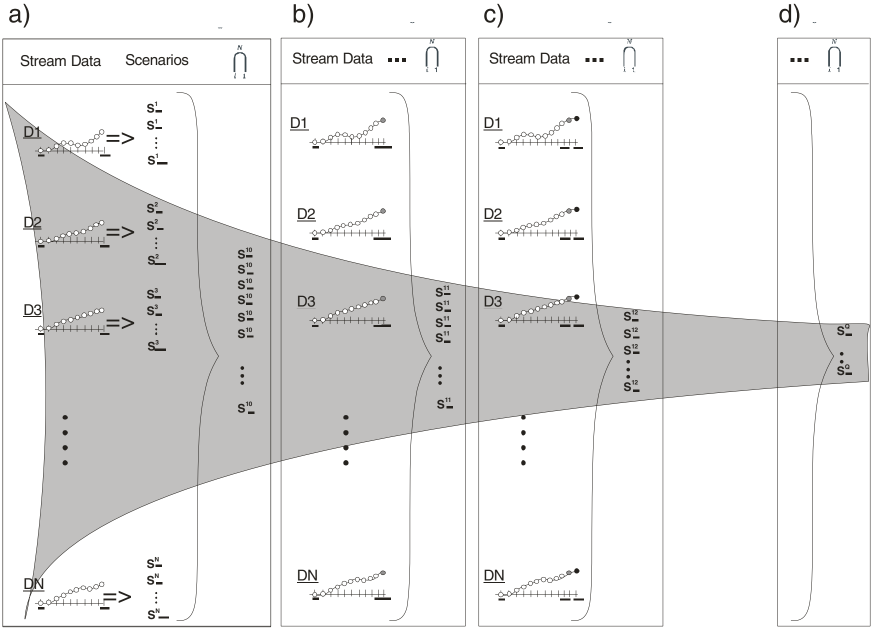

Fig. 7. Method of application of the relational database data for the analysis of streaming data

Sensors are located in same nodes of a grid used for the numerical modeling. We assumed that each sensor measures a single physical parameter probing a physical field with a constant sampling rate. The first stage of the proposed procedure is presented in part A of figure 7. The measurement system is activated in cases of flood risk, usually by the local authority responsible for crisis management. The sensors start measurements and the data is continuously transmitted to the database over the wireless transmission system. The data interpretation starts after a predetermined number of data samples have been received, for example 10. The samples are used in the scenario identification procedure. The recorded time series is compared with the synthetic time series modeled in the same place (in the same node) for different scenarios. As a result of the procedure the scenarios that give the best-fitted models are selected. The maximum misfit is selected empirically. Finally, for each sensor the set of characteristic scenarios that best reflect the dynamic processes are chosen. In the final step of the procedure the scenarios that were identified for most of sensors are recognized as scenarios common for all sensors. If we decide that only the scenarios that were identified for each sensor should be accepted we have to choose the common part of all identified scenarios (logical product). The second stage of the procedure is presented in Fig 7b. When the new sample appears on the sensor it is appended to the existing time series. Next, similarly to the previous step, the right scenarios for each sensor are identified and a common part of all the scenarios is chosen. As a result, we obtain common models, most probable for all parts of the embankment, valid for step two. The same operations are repeated in the following steps (see Fig. 7 c,d). In each step we obtain a set of permissible scenarios. Step by step the scenarios are identified with better precision, because the time series increases in length. Eventually, step by step, the number of permissible scenarios decreases. Preferably, only the scenarios that were present in the previous step should be accepted in the next one. Finally, in the last step of the procedure, the relatively small number of acceptable scenarios is identified. They optimally characterize the dynamic conditions in the examined structure. What is important is that the assessment of admissible scenarios can be evaluated for any starting time of the experiment. The number of accepted scenarios will be larger for that case because the time series is shorter. Consequently the assessment of the real embankment state will be less reliable.

5 Conclusions and the Further Work

The problem of processing stream data from a flood embankment monitoring system is presented in this article. The authors have introduced a real application for flood embankment monitoring into stream data systems, especially sensor networks. A model has been proposed for storing the data acquired from sensors (saturation, pore pressures, temperature), dedicated to data stream storing and processing. Another proposal considers a method for comparing input stream data with scenarios that assume stability of embankments as well as the scenarios that led to the violation of its stability. The key features of the proposed model (schema, scenario analysis) are included in this article. Future work involves consideration of such problems as efficient spatial indexing of data acquired from different measurement points on the grid of sensor networks. Another problem is the sizing of the sliding windows in order to perform proper scenario analysis [19,20]. This case and its related issues is a topic for further consideration. It may be also considered to use machine learning methods to assess the embankment stability. For example, support vector machines can be trained with large amount of data using the framework proposed in [13].

Acknowledgments. This work is financed by the National Centre for Research and Development (NCBiR), Poland, project PBS1/B9/18/2013 - (no 180535).

This work was partly support by the AGH - University of Science and Technology, Faculty of Geology, Geophysics and Environmental Protection, as a part of statutory project number No.11.11.140.032.

References

1. Balis, B., Kasztelnik, M., Bubak, M., Bartynski, T., Gubała, T., Nowakowski, P., Broekhuijsen, J.: The urbanflood common information space for early warning systems. Procedia Computer Science 4, 96–105 (2011)

2. Biot, M.A.: Theory of elasticity and consolidation for a porous anisotropic solid. Journal of Applied Physics 26(2), 182–185 (2004)
3. Botan, I., Derakhshan, R., Dindar, N., Haas, L., Miller, R.J., Tatbul, N.: Secret: a model for analysis of the execution semantics of stream processing systems. Proceedings of the VLDB Endowment 3(1-2), 232–243 (2010)
4. Carney, D., Çetintemel, U., Cherniack, M., Convey, C., Lee, S., Seidman, G., Stonebraker, M., Tatbul, N., Zdonik, S.: Monitoring streams: a new class of data management applications. In: Proceedings of the 28th International Conference on Very Large Data Bases, pp. 215–226. VLDB Endowment (2002)
5. Chuchro, M., Lupa, M., Pięta, A., Piórkowski, A., Leśniak, A.: A concept of time windows length selection in stream databases in the context of sensor networks monitoring. In: Bassiliades, N., Ivanovic, M., Kon-Popovska, M., Manolopoulos, Y., Palpanas, T., Trajcevski, G., Vakali, A. (eds.) New Trends in Database and Information Systems II. AISC, vol. 312, pp. 173–183. Springer, Heidelberg (2015)
6. Cranor, C., Johnson, T., Spataschek, O., Shkapenyuk, V.: Gigascope: a stream database for network applications. In: Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, pp. 647–651. ACM (2003)
7. Flak, J., Gaj, P., Tokarz, K., Wideł, S., Ziębiński, A.: Remote monitoring of geological activity of inclined regions – the concept. In: Kwiecień, A., Gaj, P., Stera, P. (eds.) CN 2009. CCIS, vol. 39, pp. 292–301. Springer, Heidelberg (2009)
8. Gaj, P., Kwiecień, B.z.: The general concept of a distributed computer system designed for monitoring rock movements. In: Kwiecień, A., Gaj, P., Stera, P. (eds.) CN 2009. CCIS, vol. 39, pp. 280–291. Springer, Heidelberg (2009)
9. Golab, L., Özsu, M.T.: Issues in data stream management. ACM Sigmod Record 32(2), 5–14 (2003)
10. Golab, L., Özsu, M.T.: Processing sliding window multi-joins in continuous queries over data streams. In: Proceedings of the 29th International Conference on Very Large Data Bases , vol. 29, pp. 500–511. VLDB Endowment (2003)
11. Gorawski, M., Gorawska, A., Pasterak, K.: Evaluation and development perspectives of stream data processing systems. In: Kwiecień, A., Gaj, P., Stera, P. (eds.) CN 2013. CCIS, vol. 370, pp. 300–311. Springer, Heidelberg (2013)
12. Itasca Consulting Group, Inc.: FLAC Fast Lagrangian Analysis of Continua and FLAC/Slope – User's Manual (2008)
13. Kawulok, M., Nalepa, J.: Support vector machines training data selection using a genetic algorithm. In: Gimel'farb, G., Hancock, E., Imiya, A., Kuijper, A., Kudo, M., Omachi, S., Windeatt, T., Yamada, K. (eds.) SSPR & SPR 2012. LNCS, vol. 7626, pp. 557–565. Springer, Heidelberg (2012)
14. Konieczny, M.: Enriching WSN environment with context information. Computer Science 13(4) (2012), `http://journals.agh.edu.pl/csci/article/view/47`
15. Krzhizhanovskaya, V.V., Shirshov, G., Melnikova, N., Belleman, R.G., Rusadi, F., Broekhuijsen, B., Gouldby, B., Lhomme, J., Balis, B., Bubak, M., et al.: Flood early warning system: design, implementation and computational modules. Procedia Computer Science 4, 106–115 (2011)
16. Madden, S.: Data management in sensor networks. In: Römer, K., Karl, H., Mattern, F. (eds.) EWSN 2006. LNCS, vol. 3868, p. 1. Springer, Heidelberg (2006)
17. Madden, S., Franklin, M.J.: Fjording the stream: An architecture for queries over streaming sensor data. In: Proceeding of18th International Conference on Data Engineering, pp. 555–566. IEEE (2002)
18. Madden, S.R., Franklin, M.J., Hellerstein, J.M., Hong, W.: TinyDB: an acquisitional query processing system for sensor networks. ACM Transactions on database systems (TODS) 30(1), 122–173 (2005)

19. Patroumpas, K., Sellis, T.: Window update patterns in stream operators. In: Grundspenkis, J., Morzy, T., Vossen, G. (eds.) ADBIS 2009. LNCS, vol. 5739, pp. 118–132. Springer, Heidelberg (2009)
20. Patroumpas, K., Sellis, T.: Subsuming multiple sliding windows for shared stream computation. In: Eder, J., Bielikova, M., Tjoa, A.M. (eds.) ADBIS 2011. LNCS, vol. 6909, pp. 56–69. Springer, Heidelberg (2011)
21. Pieta, A., Bala, J., Dwornik, M., Krawiec, K.: Stability of the levees in case of high level of the water. In: 14th SGEM Geoconference On Informatics, Geoinformatics And Remote Sensing – Conference Proceedings, vol. 1, pp. 809–815 (2014)
22. Piórkowski, A., Leśniak, A.: Using data stream management systems in the design of monitoring system for flood embankments. Studia Informatica 35(2), 297–310 (2014)
23. Płaczek, B., Bernaś, M.: Optimizing data collection for object tracking in wireless sensor networks. In: Kwiecień, A., Gaj, P., Stera, P. (eds.) CN 2013. CCIS, vol. 370, pp. 485–494. Springer, Heidelberg (2013)
24. Stonebraker, M., Çetintemel, U., Zdonik, S.: The 8 requirements of real-time stream processing. ACM SIGMOD Record 34(4), 42–47 (2005)
25. Sullivan, M.: Tribeca: A stream database manager for network traffic analysis. In: VLDB, vol. 96, p. 594 (1996)
26. Szydlo, T., Nawrocki, P., Brzoza-Woch, R., Zielinski, K.: Power aware MOM for telemetry-oriented applications using GPRS-enabled embedded devices – levee monitoring use case. In: Federated Conference on Computer Science and Information Systems (FedCSIS), September 7-10 (in print, 2014)
27. Szydlo, T., Zielinski, K.: Adaptive Enterprise Service Bus. New Generation Computing 30(2–3), 189–214 (2012)
28. Yao, Y., Gehrke, J.: The cougar approach to in-network query processing in sensor networks. ACM Sigmod Record 31(3), 9–18 (2002)
29. Zielinski, K., Szydlo, T., Szymacha, R., Kosinski, J., Kosinska, J., Jarzab, M.: Adaptive SOA Solution Stack. IEEE Transactions on Services Computing 5(2), 149–163 (2012)