Storing Synopsis of Data Events

Ayaz ul Hassan Khan (email: ahkhan@kfupm.edu.sa)
Information and Computer Science Department
King Fahd University of Petroleum and Minerals

ABSTRACT

Data streams are continuous, unbounded, rapid, time-varying streams of data elements. Analysis of the discrete events or messages flows in from different sources is critical to make wise decisions such as a stock trader wants to look at the stock trend over a period of time to make the best deal at the right time. Event Stream Processing is performed to monitor streams of event data, analyze those events and act upon opportunities. NEsper is the leading open source Event Stream processing solution which works on a continuous execution model allowing applications to store queries and run the data through. NEsper engine provides responses as events in real-time whenever a condition occur as per the user-defined queries. The focus of this paper is to discuss summary generation and storage of streams so that it can provide patterns and trend analysis. Stock event streams are sampled, summarized and stored for trend analysis. Random reservoir sampling is used for generating samples.

BACKGROUND

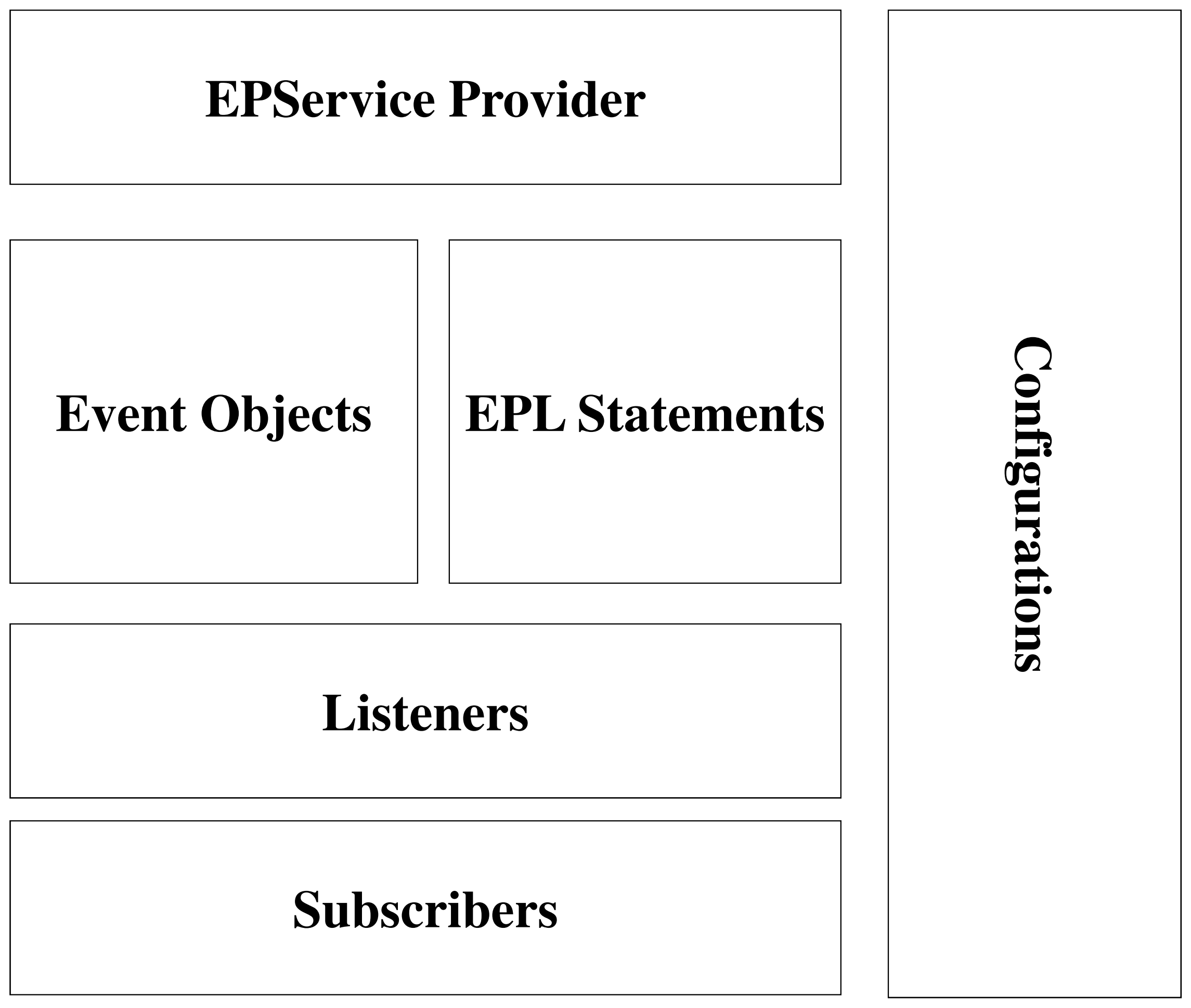
NESPER – the CEP Engine

Complex Event Processing or CEP includes event data analysis, emphasizes on patterns of events and abstracting and simplifying information in the patterns that flow between information systems to gain valuable information in real-time. NEsper [6] is a real time engine that has been developed to address the requirements of applications that analyze and react to events. It triggers actions when event conditions occur among event streams.

“Event Driven Architecture (EDA) is a software architecture supporting the production, detection, consumption of and reaction to events.” [6]
Although NEsper cannot be classified as a DSMS, yet CEP is a vital part of a DSMS and implementing it is a good idea for understanding all the concepts and areas described above.

Querying Events in NEsper

- Event Stream Analysis
- Event Pattern Matching
- Combined Pattern Matching with Event Stream Analysis



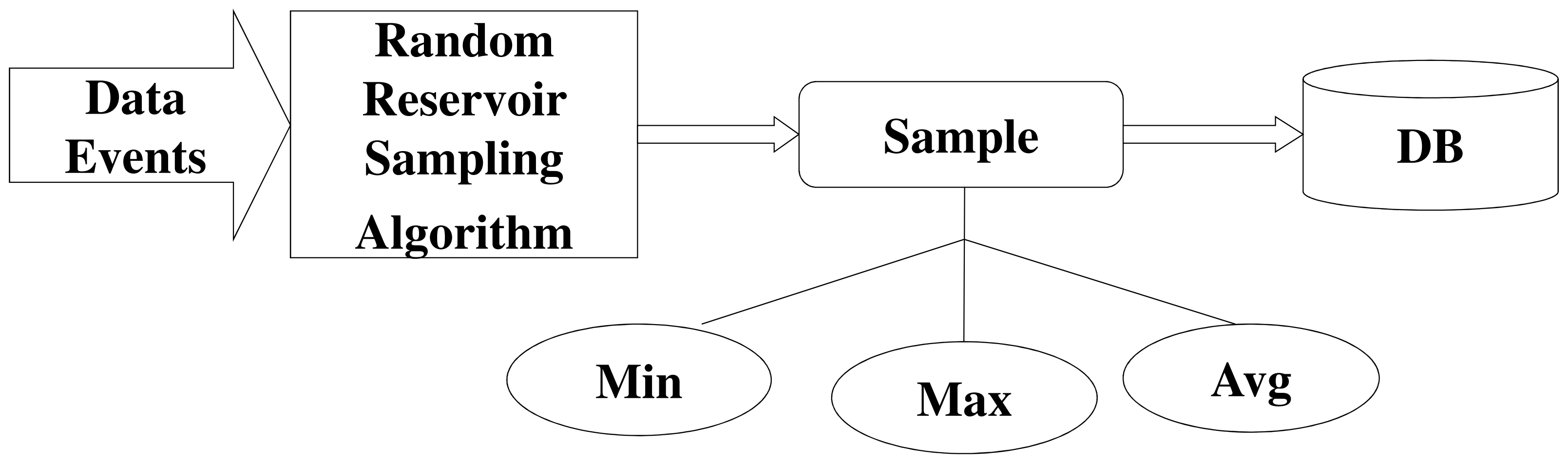
NEsper-(Event Driven) Architecture

PROPOSED APPROACH

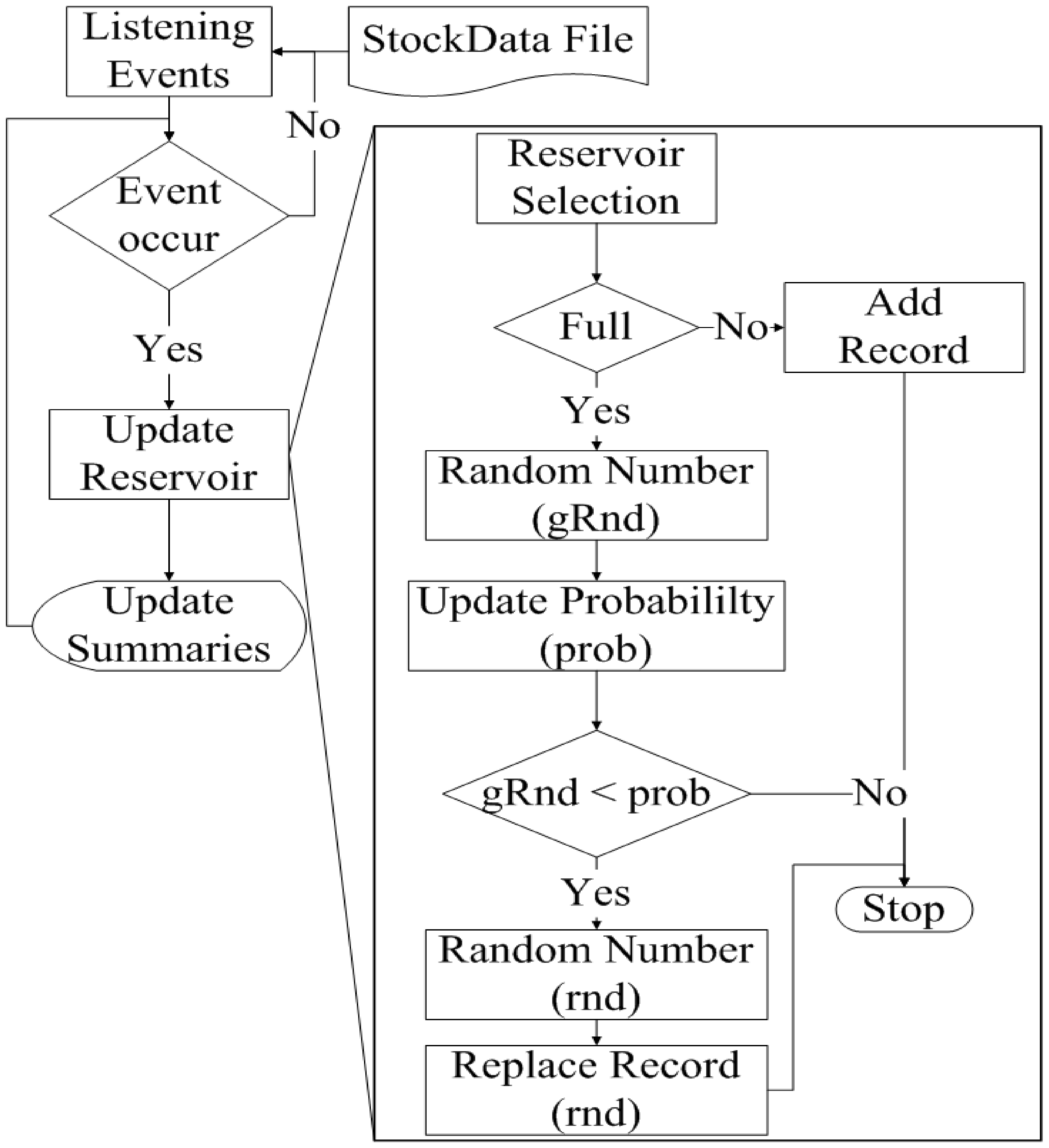
Our proposed solution of generating samples from stream and storing them is implemented on Stock readings. Stock data arrives in form of streams and Complex Event Processing is performed on it. NEsper is used for event processing, it queries the stream and stock data is exhibited to the user. The user has the option to select the companies of his choice to view their data. For instance, if the user selects four company symbols, the data of those companies is extracted from the stream and is displayed to the user.

Reservoirs will be formed as the user will select a company from the list. As the user will add company’s symbol reservoirs will be created and the sample will be produced through Random Reservoir Sampling. Sample of each company will be stored in its respective reservoir in the database. Storing the synopsis of the stream will help in maintaining history, trend analysis and pattern mining.

The stored sample is further used to provide summary to the user that is minimum, maximum and average values of the stock readings which he can see in real time.



Sample Aggregation

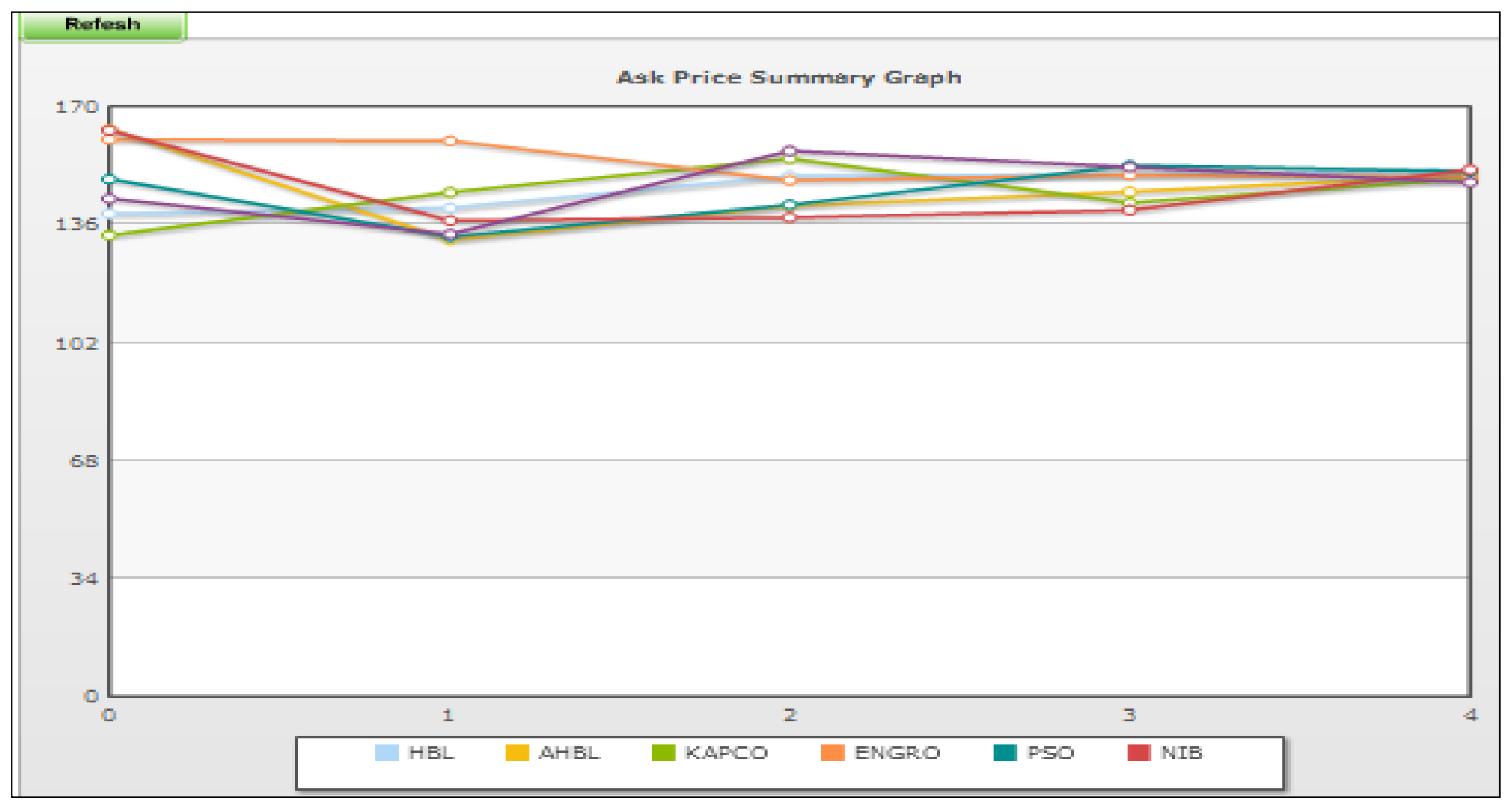


Overall processing of data stream

RESULTS

Displaying graphs
Real time graphs are displayed to the user through which he can view the progress rate of the desired companies. By the help of the chart the user can easily appreciate each symbol’s trend immediately.

Accuracy
As all the analysis is done in the sampled data its accuracy is very crucial. The data in the reservoir represents the sample of a stream so it should be accurate. To validate the precision of the sample we compare the average values of the sampled data with the average values of the stream. This will verify the accuracy of the Random Reservoir Sampling algorithm as well as the analysis done. **Random Reservoir Algorithm is on an average 97.8% accurate according to these randomly chosen count values.**



CONCLUSIONS AND FUTURE WORKS

Data streams have significant importance and researches are carried out in making them more useful and informative. Adding persistence in stream makes it more efficient and valuable for the user. Getting trend analysis and pattern mining in real time gives him a competitive advantage and he can take critical decisions in less time. Summary of the stream gives a glance of the arrived stream in just time. The user will also be able to view the history and reports of the particular time interval.

Our future work encompasses putting filters in NEsper such as on AskPrice, Volume etc. These filters will exhibit the stream which matches with conditions given by the user only. Furthermore, a data presentation layer can be added to support multiple data formats. We need to do more experiments with different data sets to evaluate the performance and accuracy of our implementation. We will also work on finding optimal reservoir size to get best accuracy.

REFERENCES

[1] Jeffry Scott Vitter, “Random Sampling with a Reservoir”, ACM Transactions on Mathematical Software, Vol. 11, No. 1, March 1985, pg 37-57.
[2] Manoranjan Dash, Willie Nj, “Efficient Reservoir Sampling for Transactional Data Streams”, Sixth IEEE International Conference on Data Mining, 2006.
[3] Hiroyuki Kitagawa, Yousuke Watanabe, “Stream Data Management Based on Integration of a Stream Processing Engine and Databases”, IFIP International Conference on Network and Parallel Computing, 2007
[4] S. Muthukrishnan, “Data Streams: Algorithms and Applications”, now Publisher Inc., 2005.
[5] “Reservoir Sampling”, http://www.itl.nist.gov/div897/sqg/dads/HTML/reservoirSampling.html
[6] “Event Stream Intelligence with Esper and NEsper” and “Tutorial”, esper.codehaus.org