

Event Processing – past, present and future

Opher Etzion

IBM Haifa Research Lab

opher@il.ibm.com

ABSTRACT

Analysts have marked *Event Processing* as the most growing segment in enterprise computing during years 2008 and 2009, furthermore, this trend is expected to continue. Many of the large and medium software companies (IBM, Oracle, Microsoft, Sybase, Progress Software, Software AG and TIBCO) are now offering event processing products as well as a collection of smaller companies. Other indications for the emerging nature of this area are: extensive coverage by analysts as a separate area, the establishment of a dedicated research community with an annual conference (DEBS), and the establishment of a consortium that includes vendors and academic people as EPTS (Event Processing Technical Society) <http://www.ep-ts.com/>.

The early event processing commercial products were mostly descendents of research projects rooted in multiple disciplines, some of them are data management disciplines (active databases, data stream management, temporal databases) and some are rooted in other areas (discrete event simulation, distributed computing, formal verification).

The tutorial is intended for a technical audience that is interested in deep dive into understanding event processing. The audience will gain insights about event processing: What it really means? Where does it come from? How does it relate to research concepts (e.g. stream computing) as well as enterprise computing terms (e.g. Business Rules Management Systems)? The audience will also gain insights into the current state of the art, the leading architectures, the basic building blocks of event processing, and the various programming styles exemplified by code examples. Last but not least, the audience will gain insights about the current trends, and the research challenges that exist, this part will be based on the discussions in the Event Processing Dagstuhl seminar that was held in May 2010 <http://www.dagstuhl.de/de/programm/kalender/semhp/?semnr=10201>. The tutorial slides are available for public viewing on slideshare <http://www.slideshare.net/search/slideshow?q=opher+etzion+%2B+vldb2010+tutorial>

The current generation of event processing products [14] has been preceded by several research projects in the 1990-ies: Rapide in Stanford [15], Infospheres in Cal Tech [2], Apama in Cambridge University [6] and Amit in IBM Haifa Research Lab [1]. In later phase there were some research project that have taken the stream oriented approach such as the Stanford Stream project [3] and

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Articles from this volume were presented at The 36th International Conference on Very Large Data Bases, September 13-17, 2010, Singapore.

Proceedings of the VLDB Endowment, Vol. 3, No. 2

© 2010 VLDB Endowment 2150-8097/10/09... \$10.00

Aurora [5].

A collection of start-up companies, many of them descendents of these projects have emerged, some survived, and some did not. From those who survived some were acquired by bigger companies.

Event processing as a research discipline has multiple ancestors. In the database area it is a descendent of work done in active databases [19], temporal databases [12], and data stream management systems [9]. Other ancestors are inference rules, discrete event simulation, and distributed computing (pub/sub).

1. Event processing – present

In the commercial domain, all the bigger software vendors own products in the event processing areas, getting it to the main stream of enterprise computing. Some books describing the business side [8] and the technical side of event processing [10] have emerged. There is work to define common conceptual model of event processing based on event driven architectures [18] and event processing networks [10]. The current work is centered around: architectures [17], languages [7], engineering aspects such as: scalability and optimization [13], security [4], pattern matching [10], context based event processing [11], temporal and spatio-temporal event processing. The tutorial surveys each of these aspects.

2. Event processing – future

We discuss six trends identified in event processing applications:

1. Going from more specialized to wider platforms that need to support variety of functional and non-functional requirements.
2. Going from monolithic implementations to diversified implementations.
3. Going from proprietary to standard-based implementations.
4. Going from programming based development to semi-technical developers,
5. Going from stand-alone event processing to embedded solution within larger platforms
6. Going from reactive applications to proactive applications

The tutorial explains each of these aspects and surveys research challenges in the usability, engineering, semantic and software engineering aspects, as well as challenges of use in biological and socio-technical systems.

3. ACKNOWLEDGMENTS

Many people have contributed to the tutorial content.

Most notable, my partner in authoring the "Event Processing in Action" book [10] Peter Niblett, and my colleagues from IBM Haifa Research Lab, especially: Ella Rabinovich and Guy Sharon.

4. REFERENCES

- [1] Asaf Adi, Opher Etzion: Amit - the situation manager. VLDB J. 13(2): 177-203 (2004)
- [2] Elli Albek, Eric Bax, Greg Billock, K. Mani Chandy, Ian Swett: An Event Processing Language (EPL) for Building Sense and Respond Applications. IPDPS 2005
- [3] Arvind Arasu, Brian Babcock, Shivnath Babu, Mayur Datar, Keith Ito, Itaru Nishizawa, Justin Rosenstein, Jennifer Widom: STREAM: The Stanford Stream Data Manager. SIGMOD Conference 2003: 665
- [4] Jean Bacon, David M. Eysers, Jatinder Singh, Brian Shand, Matteo Migliavacca, Peter R. Pietzuch: Security in Multi-domain Event-based Systems. *it - Information Technology* 51(5): 277-284 (2009)
- [5] Hari Balakrishnan, Magdalena Balazinska, Donald Carney, Ugur Çetintemel, Mitch Cherniack, Christian Convey, Eduardo F. Galvez, Jon Salz, Michael Stonebraker, Nesime Tatbul, Richard Tibbetts, Stanley B. Zdonik: Retrospective on Aurora. VLDB J. 13(4): 370-383 (2004)
- [6] John Bates, Jean Bacon, Ken Moody, Mark D. Spiteri: Using events for the scalable federation of heterogeneous components. ACM SIGOPS European Workshop 1998: 58-65
- [7] Francois Bry, Michael Eckert, Opher Etzion, Jon Riecke, Adrian Paschke. Event Processing Languages. Tutorial in DEBS 2009.
- [8] K. Mani Chandy, W. Roy Schulte: Event Processing: Designing IT Systems for Agile Companies. McGraw Hill, 2009.
- [9] Sharma Chakravarthy, Qingchun Jiang: Stream data Processing: A Quality of Service Perspective. Springer 2009.
- [10] Opher Etzion, Peter Niblett: Event Processing in Action, Manning Publications 2010
- [11] Opher Etzion, Ella Rabinovich. Context based computing and context-based event processing. Tutorial in DEBS 2010. <http://www.slideshare.net/opher.etzion/debs-2010-context-tutorial>
- [12] Opher Etzion, Sushil Jajodia, Surya Sripada (eds) – Temporal databases: research and practice - Springer, 1998.
- [13] Geetika T. Lakshmanan, Yuri G. Rabinovich, Opher Etzion: A stratified approach for supporting high throughput event processing applications. DEBS 2009
- [14] Neal Leavitt, "Complex-Event Processing Poised for Growth," *Computer*, vol. 42, no. 4, pp. 17-20, Apr. 2009,
- [15] David C. Luckham, James Vera: An Event-Based Architecture Definition Language. *IEEE Trans. Software Eng.* 21(9): 717-734 (1995)
- [16] David Luckham: The Power of Events. Addison Wesley, 2002.
- [17] Adrian Paschke, Paul Vincent. Event processing reference architectures. Tutorial in DEBS 2010
- [18] Hugh Taylor, Angela Yochem, Les Phillips, Frank Martinez: Event-Driven Architecture. Addison-Wesley 2009
- [19] Jennifer Widom, Active Database Systems: Triggers and Rules for Advanced Database Processing, Morgan Kaufmann, 1995.