Formal Analysis of Complex Event Processing - Potential and Challenges

Matthias Weidlich

Deparment of Computer Science, Humboldt-Universität zu Berlin
matthias.weidlich@hu-berlin.de

Extended Abstract

Complex event processing (CEP) emerged as a paradigm to build systems that react to *situations of interest* [1]. By evaluating continuous queries over streams of events, CEP systems provide the foundation for re-active and pro-active applications in domains such as healthcare and urban transportation. Models for CEP are an active area of research and various languages for the definition of event stream queries have, so far, been proposed. While each of them provides a different syntax and semantics, they typically adopt point-based event semantics (i.e., the occurrence of an event is atomic), an attribute-based data model (an event carries payload, which is structured as key-value pairs), and comprise a set of common query operators [7], such as *sequencing* of events in terms of their temporal order; *negation* to check for the absence of an event; and *windows* to bound the temporal interval in which events are considered relevant to the query.

Common CEP applications face high-velocity event streams, which renders the evaluation of queries a performance bottleneck. Therefore, various algorithms and architectures for efficient CEP have been proposed in recent years, including techniques for parallelisation and distribution of query evaluation [3], semantic query rewriting [6], or sub-pattern sharing [4]. Moreover, reflecting on the challenges induced by distributed event sources, techniques to achieve robustness of CEP against out-of-order arrivals of events have been developed [2].

Despite all these advancements, we argue that most of these techniques adopt a pragmatic view—they strive for a technical solution of the issues as they emerge in a specific application. We therefore advocate the design of formal methods to guide the design and implementation of CEP applications. Specifically, we suggest to rely on well-established formalisms for concurrent systems to reason about the following aspects of CEP applications:

- *Query verification:* Formal analysis may reveal whether a query deployed in a CEP system can match at all, specifically if event streams satisfy domain-specific constraints.
- *Sound parallelisation:* Formal analysis may identify non-determinism in query evaluation that is introduced through paralelisation schemes for queries that potentially interact with each other.
- *Robustness guarantees:* Formal analysis may enable conclusions on the errors introduced by out-of-order event arrivals at a CEP system, thereby giving robustness guarantees.

In the light of the above reasoning tasks, we developed a formal model of CEP applications that is grounded in Petri-nets [5]. The choice of this formalism is motivated by their concurrent and local semantics, along with the broad availability of analysis algorithms and tool support.

Major challenges in this endeavour, however, have been the integration of the various perspectives of a CEP application in a single model: It requires a comprehensive formalisation of the semantics of event streams, event queries, and evaluation architectures, properly capturing their interplay. For instance, it is not sufficient to simply capture that an event occurred, but several modalities have to be encoded on the formal level: An event may have occurred, but may be consumed by a match of an event query and, thus, no longer be available to construct further matches. In such a case, the interplay with common evaluation architectures needs to be taken into account. If an event can be consumed only by a single match of a query, sequential evaluation of various match candidates of a query may lead to a different result compared to their concurrent evaluation.

Using our formal model, we are able to approach reasoning on the aforementioned aspects of CEP applications through standard reachability analysis. The question of whether a specific query can match translates into the common problem of identifying whether a specific state (i.e., a marking in the Petri-net) can be obtained from the initial state. Based thereon, conclusions can be drawn on the general possibility of generating matches; on the changes in the sets of matches obtained under different parallelisation schemes; and on the implications of out-of-order arrivals of events, whether they potentially lead to false positives and false negatives in query evaluation.

References

1. Cugola, G., Margara, A.: Processing flows of information: From data stream to complex event processing. ACM Comput. Surv. **44**(3), 15:1–15:62 (2012).
2. Liu, M., Li, M., Golovnya, D., Rundensteiner, E.A., Claypool, K.T.: Sequence pattern query processing over out-of-order event streams. In: ICDE. IEEE (2009).
3. Mayer, R., Slo, A., Tariq, M.A., Rothermel, K., Gräber, M., Ramachandran, U.: SPECTRE: supporting consumption policies in window-based parallel complex event processing. In: Middleware. pp. 161–173. ACM (2017).
4. Ray, M., Lei, C., Rundensteiner, E.A.: Scalable pattern sharing on event streams. In: SIGMOD. pp. 495–510. ACM (2016).
5. Reisig, W.: Understanding Petri Nets - Modeling Techniques, Analysis Methods, Case Studies. Springer (2013).
6. Weidlich, M., Ziekow, H., Gal, A., Mendling, J., Weske, M.: Optimizing event pattern matching using business process models. IEEE Trans. Knowl. Data Eng. **26**(11), 2759–2773 (2014).
7. Zhang, H., Diao, Y., Immerman, N.: On complexity and optimization of expensive queries in complex event processing. In: SIGMOD. pp. 217–228. ACM (2014).