Student Research Abstract: A Liquid Software-driven Semantic Complex Event Processing-based platform for health monitoring

Francesco Nocera
Polytechnic University of Bari
Bari, Italy
francesco.nocera@poliba.it

ACM Reference Format:
Francesco Nocera. 2018. Student Research Abstract: A Liquid Software-driven Semantic Complex Event Processing-based platform for health monitoring. In *SAC 2018: SAC 2018: Symposium on Applied Computing , April 9–13, 2018, Pau, France.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3167132.3167454

1 INTRODUCTION

Technological innovation is driving profound changes in all productive fields. Consequently it is contributing to a new definition of the organizations and processes, which increasingly require new skills and responsibilities profiles. This path has also involved the health field where technologies have taken a leading role, it's becoming an integral part of the health service and they are establishing more and more inseparable interconnections between health and technology. The progress in medical technology is driving a continuous improvement of health and predictive diagnostic and therapeutic outcomes. Throughout the industry that deals with health, but especially in hospitals, the significant presence of technology, must ensure safe and appropriate use in various phase of prevention, diagnosis and treatment. It is expected that the Healthcare Industrial IoT (HealthIIoT) will be one of the main players in the Industrial Internet of Things (IIoT)-driven healthcare industry. IIoT has had a remarkable influence across many large and small healthcare industries. As a result, an increasing number of wearable IoT devices, tools, and apps are being used for different monitoring applications (e.g., glucose monitors, ECG monitors, and blood pressure monitors). According to G. Cugola and A. Margara [3] the concepts of timeliness and flow processing are crucial for justifying the need for a new class of systems. Complex event Processing (CEP) is one of these emerging models to monitor and react to continuously arriving events in real-time or near-real. Critical factors for real-time event-based and Embedded Systems are event detection and the enormous amount of information available on the events. The continuous streams of high-level events require real-time intelligent processors. Human knowledge domain will greatly affect the decision making support system. Knowledge representation is the method used to encode knowledge in an intelligent systems knowledge base. Events-based semantic models can improve the quality of event processing by using metadata in combination with knowledge bases consisting of ontologies and rules.

SAC 2018, April 9–13, 2018, Pau, France

ACM ISBN 978-1-4503-5191-1/18/04.
https://doi.org/10.1145/3167132.3167454

2 BACKGROUND AND RELATED WORK

Currently, an increasing number of distributed applications requires continuous analysis of flows of data and real-time response to complex queries. Furthermore, it is very important to design the overall architecture [9] and to decide the way data should be stored because this choice will determine later also the way in which data will be extracted. Today ontologies are widely used to model and encode domain's knowledge and allow us to reason about this knowledge . The concept of Complex Event Processing (CEP) was introduced by David Luckham in his seminal work [6]. CEP systems can be classified as Advanced Decision Support Systems (ADSS). The key characteristic of a CEP system is its capability to handle complex event situations, detecting patterns, creating correlations, aggregating events and making use of time windows. The Liquid Software metaphor [5] refers to software that can operate seamlessly across multiple devices owned by one or multiple users. Liquid Software applications can take advantage of the computing, storage and communication resources available on all the devices owned by the user. Liquid Software applications can also dynamically migrate from one device to another, following the user's attention and usage context. The key design goal in Liquid Software development is to minimize the additional efforts arising from multiple device ownership (e.g., installation, synchronization and general maintenance of personal computers, smartphones, tablets, home and car displays, and wearable devices), while keeping the users in full control of their devices, applications and data. A complete Systematic Literature Review about Application of Complex Event Processing Techniques and architecture to Big Data Related to Healthcare has been published [1].

3 APPROACH AND UNIQUENESS

The aim of this paper will be the design of an integrated platform consisting of integrated components designed to remotely manage the paths of prevention, diagnosis and treatment. In order to support the different levels of prevention, the platform should provide a monitoring system integrated with a module of management of clinic compliance (care process), the real-time management of the events by a CEP system and a connection system to the own caregiver or physician for the management of teleconsultation (diagnosis process) according to Liquid Software architecture and The process of care management involves the management of compliance and must include the link to the medical plan. Also, it must provide the delivery of care using any devices in the patient's home.

In particular, all scenarios include a system of systematic data collection and their organization in a knowledge base used by decision makers at all levels and the definition of dashboards for the governance of health care. A health monitor is a personalized system that allows a person and their caregivers to monitor the person's health status. Health monitors may be particularly useful for chronically ill people as well as for elderly citizens. The data may be captured from sensors and devices at the person as well as from stationary sensors in their home or in a specific clinical area for the atmosphere data collection. Alarms are set up to alert the person and, if necessary, doctors or a remote caregiver. Sensors automatically capture personalized health data such as heart rate, blood pressure, respiration rate, ECG, oxygen saturation in blood, the location of the person in reference to a room [7]. In addition, specific regular measurements for particular health conditions are performed by the person and the results are filtered for emergency situations as well as kept for long term observation. Health monitors offer many challenges derived from the high volume of low level events and the need to derive higher level events that must be propagated. Sources include real-time data by sensors, medical devices, data stream generate real-time data that are captured by other systems for continuous monitoring, together with patient's historical data (Digital medical record, Electronic health record). Depending on the scenario, anomalies may be resolved by automated responses or alerts for human or machine intervention. CEP extends this capability by correlating multiple events through a common interface that invokes an embedded rules engine. Event filtering evaluates a specified logical condition based on event attributes, and, if the condition is true, publishes the event to the destination stream as a notification or alert. The main challenge is the huge amount of data collection and their organization in a knowledge base. Ontologies play an important key role in the knowledge-based CEP [4, 8]. They cover the conceptualization of the application domain to allow reasoning on events and other non-event concepts. The event processing domain is described by a modular and layered ontology model which can be reused in different scenario application. Important general concepts such as event, action, situation, space/place, time, agent and process are defined based on meta-models and pluggable ontologies which are in a modularized ontological top-level structure. These general concepts defined in the top-level ontologies can be further specialized with existing domain ontologies and ontologies for generic tasks and activities. Indeed has been implemented a component able to discover business processes from the generated event log and a suitable level of abstraction. The component is a support for process monitoring and analysis. The proposed components extend CNMining PROM plugin for automated discovery of processes, and builds an UML Activity Diagram abstracting the process resources and the role resources [2]. The assignment of the resources and corresponding roles to the activity are modeled by abstraction on a Domain and Business Ontologies. The overall described architecture is depicted in Figure 1.

4 RESULTS AND CONTRIBUTIONS

The real-time platform modelled integrates Complex Event Processing and semantic web technologies and approaches. Together, these components create an agile, high performance, scalable platform

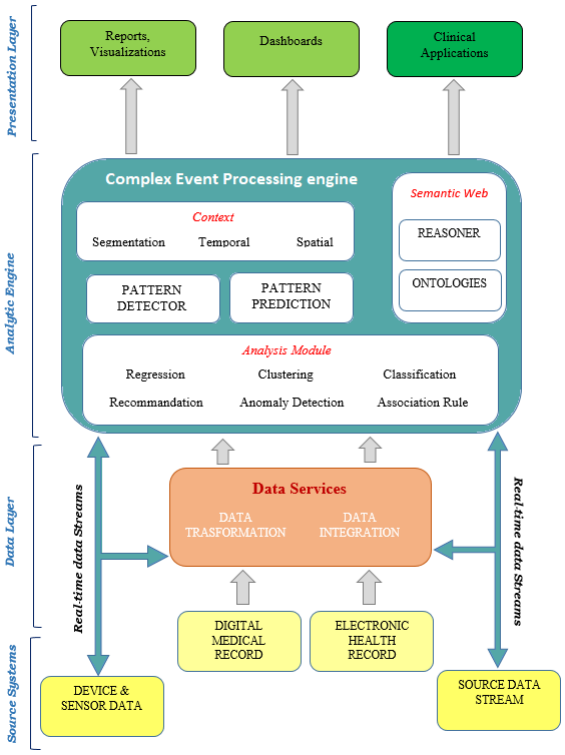

Figure 1: Proposed functional Architecture.

that can deliver fast insights through real-time queries, pattern matching and anomaly detection, continuous analytics and triggers notifications and alerts based on a CEP engine around a Liquid software architecture.

REFERENCES

[1] M. Atzmueller. *Enterprise Big Data Engineering, Analytics, and Management.* IGI Global, 2016.
[2] S. Bistarelli, T. D. Noia, M. Mongiello, and F. Nocera. Pronto: an ontology driven business process mining tool. *Procedia Computer Science*, 112(Supplement C):306 – 315, 2017. Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 21st International Conference, KES-20176-8 September 2017, Marseille, France.
[3] G. Cugola and A. Margara. Processing flows of information: From data stream to complex event processing. *ACM Comput. Surv.*, 44(3):15:1–15:62, June 2012.
[4] T. Di Noia, M. Mongiello, F. Nocera, and E. Di Sciascio. Ontology-based reflective iot middleware-enabled agriculture decision support system.
[5] A. Gallidabino, C. Pautasso, V. Ilvonen, T. Mikkonen, K. Systä, J. Voutilainen, and A. Taivalsaari. Architecting liquid software. *Journal of Web Engineering*, 16(5&6):433–470, 2017.
[6] D. Luckham. *The power of events*, volume 204. Addison-Wesley Reading, 2002.
[7] M. Mongiello, T. di Noia, F. Nocera, E. di Sciascio, and A. Parchitelli. Context-aware design of reflective middleware in the internet of everything. In P. Milazzo, D. Varró, and M. Wimmer, editors, *Software Technologies: Applications and Foundations: STAF 2016 Collocated Workshops: DataMod, GCM, HOFM, MELO, SEMS, VeryComp, Vienna Austria, July 4-8, 2016, Revised Selected Papers*, pages 423–435, Cham, 2016. Springer International Publishing.
[8] M. Mongiello, L. Patrono, T. Di Noia, F. Nocera, A. Parchitelli, I. Sergi, and P. Rametta. A complex event processing based smart aid system for fire and danger management. In *Advances in Sensors and Interfaces (IWASI), 2017 7th IEEE International Workshop on*, pages 44–49. IEEE, 2017.
[9] F. Nocera. Fuzzy ontology-driven web-based framework for supporting architectural design: Student research abstract. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, SAC '16, pages 1361–1362, New York, NY, USA, 2016. ACM.