A Data Streams Processing Platform for Matching Information Demand and Data Supply

Jānis Grabis(✉), Jānis Kampars, Krišjānis Pinka, and Jānis Pekša

Institute of Information Technology, Riga Technical University,
Kalku 1, Riga 1658, Latvia
{grabis, janis.kampars, krisjanis.pinka,
janis.peksa}@rtu.lv

Abstract. Data-driven applications are adapted according to their execution context, and a variety of live data is available to evaluate this contextual information. The BaSeCaaS platform described in this demo paper provides data streaming and adaptation services to the data driven applications. The main features of the platform are separation of information requirements from data supply, model-driven configuration of data streaming services and horizontal scalable infrastructure. The paper describes conceptual foundations of the platform as well as design of data stream processing solutions where matching between information demand and data supply takes please. Light-weight open-source technologies are used to implement the platform. Application of the platform is demonstrated using a winter road maintenance case. The case is characterized by variety of data sources and the need for quick reaction to changes in context.

Keywords: Data stream · Adaptation · Context · Model-driven

1 Introduction

Data-driven applications (DDA) rely on data availability and intelligent processing to guide their execution. These applications have certain information demands, which can be formally described as their execution context. On the other hand modern information and communication technologies provide ample opportunities for data capture though organizations often struggle with applying these data [1], especially if different types of external data are used. The external data are often characterized by high level of volatility and lack of meta-information about their content and usefulness. The organizations might know their information needs while it is difficult to identify appropriate data sources and to transform data in a suitable form [2].

The Capability Driven Development methodology [3] addresses the aforementioned challenges of developing DDA. It provides methods and guidance to develop context-aware and adaptive applications. Computational tools have been developed to support the methodology. The BaSeCaaS platform described in this demo paper furthers development of tools specifically dealing with processing of data streams for

C. Cappiello and M. Ruiz (Eds.): CAiSE Forum 2019, LNBIP 350, pp. 111–119, 2019.
https://doi.org/10.1007/978-3-030-21297-1_10

needs of DDA. It is intended for application cases characterized by: (1) variety of stakeholders (i.e., data suppliers and information consumers); (2) distributed, volatile and heterogeneous data sources; (3) high volume data streams; (4) computationally demanding application adaptation algorithms; and (5) near real-time response. The main features of the BaSeCaaS are: (1) model-based specification of data streams processing requirements; (2) automated deployment of horizontally scalable data streams processing environment; (3) separation of information requirements and data sources; and (4) decoupling of computationally intensive DDAs adaptation logics from the core business logics.

The survey [4] of more than 30 data stream processing tools reveals that the field is mature though there is strong emphasis on using push based processing, processing languages have variable expressiveness and topology of systems is a concern of further research. Another survey [5] identifies requirements towards streaming tools and points out that processing language and historical/current data integration are two major limitations. These tools mainly address technical concerns while BaSeCaaS focuses on making data stream processing accessible to consumers. Similar concerns are addressed in [6] proposing a knowledge based approach to deal with data heterogeneity. Data markets and platforms [7] dealing with matching data consumers and providers are relatively early stages of development for data streaming applications.

The objective of these demo paper is to describe the overall design of BaSeCaaS and to demonstrate its application case. The platform supports model-driven development and it consists of three main layers, namely, modeling, service and infrastructure layers. The modeling layer is responsible for representing the data processing problem, the service layer implements data processing services and the infrastructure layer provides scalable computational resources of execution of the services. The platform is implemented using a set of open-source lightweight technologies. The application case considered deals with the winter road maintenance problem.

The rest of the paper is structured as follows. Section 2 describes conceptual foundations of the BaSeCaaS. The platform's application scenario is discussed in Sect. 3. Technological solutions are presented in Sect. 4 and a brief demonstration is provides in Sect. 5. Section 6 concludes.

2 Foundations

The platform is developed on the basis of the Capability Driven Development metamodel [3] and focuses on parts dealing with context processing. Figure 1 shows its key concepts. The information demand is defined using context elements (CE), where context characterizes DDA execution circumstance. The context elements are thought to have a clear business interpretation (i.e., they drive execution of the DDA). Adjustments define actions to be performed as a result of changes in the context what can be perceived as context dependent adaptation of the application. Raw observations of DDA execution circumstances are referred as to measurable properties (MP). They specify available data and often there is no clear idea about their usage. Context providers for MP are physical endpoints of the context acquisition channel in BaSeCaaS. They are used by providers to post context observations.

Relations among CE and MP are established using Context Calculation element, which specifies transformation of raw data streams into useful information. This way raw observations are decoupled from information consumption and the context calculations can be modified depending on data availability and other considerations. In the boundary case without available data providers, the DDA becomes a static application. Similarly, the Adjustment trigger binds adjustments with context. The adjustment is enacted if context elements assume specific values. The adjustments also use the context elements to determine an appropriate action.

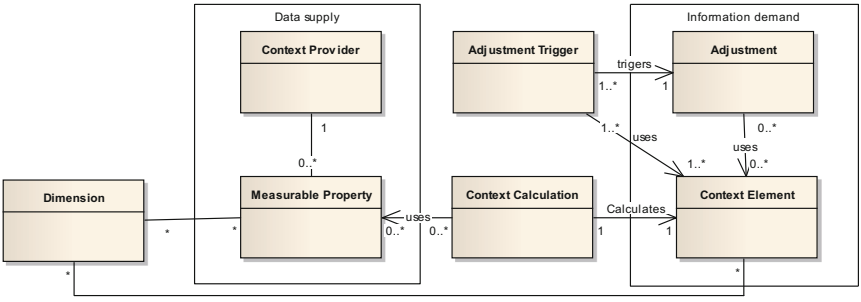

Fig. 1. Key concepts used in the BaSeCaaS platform

The data stream processing problem is tackled in two phases: (1) design time; and (2) run time. The design time phase (Fig. 2) deals with definition of the data processing problem following a model-driven approach. On the demand side, information requirements are represented as context elements and context-aware adaptation of DDA

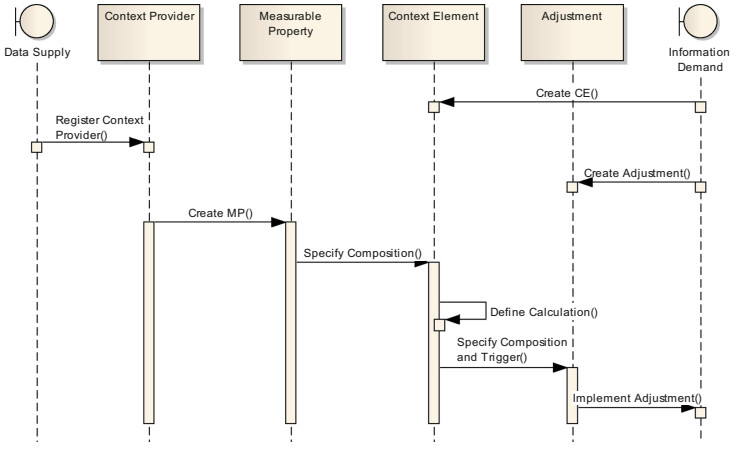

Fig. 2. The design phase of data streams processing

is specified using adjustments. On the supply side, context data providers are registered with the platform and appropriate MP are specified. As indicated above, the demand and supply sides are linked via context calculations and MP used to calculate CE are specified and joint together by creating context element composition (i.e., rules for joining multiple streams). In the context calculation, the MP are filtered by using sliding window and aggregation functions. The context calculations can be specified in a form of rules or arbitrary calculations. The adjustments implement the adaptation logics of data driven applications and passes decisions made onto these applications via their interfaces. That allows of target application independent modification and execution of computationally demanding and volatile adaptive functions.

In the run-time phase, context providers post context measurements in the BaSeCaaS via its API. The orchestration service validates the data received and archives data for batch analysis. It makes data available for context element value calculation. Similarly, the calculated context element values are made available to the Adjustment trigger and Adjustment execution services. The Adjustment trigger service uses the context element values to determine whether an adjustment should be invoked and passes this message to the Orchestration service. The Adjustment execution service gets notification from the Orchestration service to execute the adjustment as well as the context element values necessary for its evaluation. The adjustment logics is evaluated and appropriate functions of the data driven application are invoked using the adjustment evaluation results as inputs (Fig. 3).

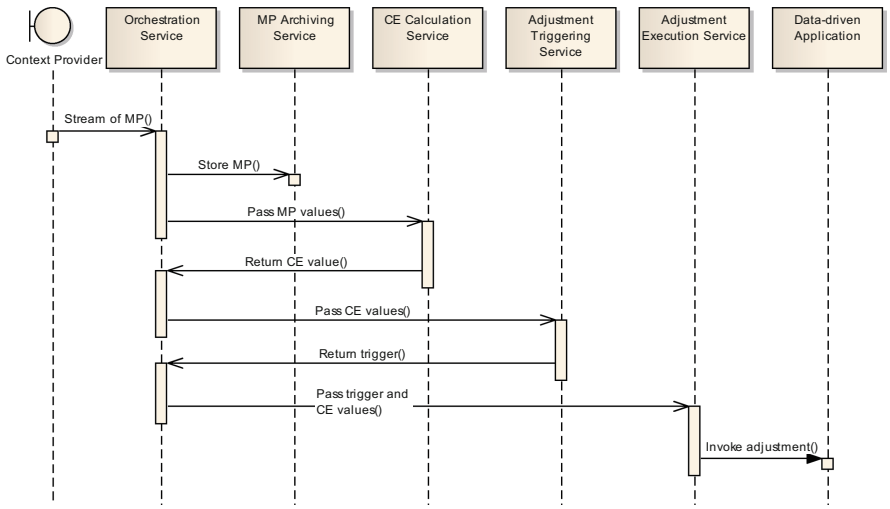

Fig. 3. The run-time phase of the data streams processing

3 Scenario

The application case considered is winter road maintenance (WRM) [8]. The case is characterized by the need for timely reaction to changes in road conditions due to snow and icing. The delayed action might cause traffic accidents with severe consequences. Additionally, there are many information consumers ranging from road maintenance and emergency services to drivers. From the data supply perspective, there is a diversity of data sources. The bulk of data are provided by field surveying, which has relatively low frequency and varying coverage. There are road monitoring weather-stations and cameras operated by different entities. Their data are comprehensive though limited to major sections of the road network. In the case of insufficient coverage, non-conventional data sources such as mobile fleet and crowd-sourced data.

As an example the following information needs are considered:

- Road conditions – road conditions characterize visual appearance of the road with possible values Bare, Partly Covered and Covered;
- Driving conditions – forecasting driving conditions with values Good, Fair, Caution and Poor;
- Recommended speed – depending on road conditions speed limit is changed;
- Snow removal needs – the level of urgency of the snow removal.

The adjustments defined are:

- Snow removal prompt – depending on snow removal needs a road maintenance company receives a notification;
- Change recommended speed – depending on recommended speed information is changed in smart road signs;
- Road conditions warning – depending on road conditions, on-line road maps are updated;
- Driving conditions warning – depending on driving conditions notification to drivers and other stakeholders are provided.

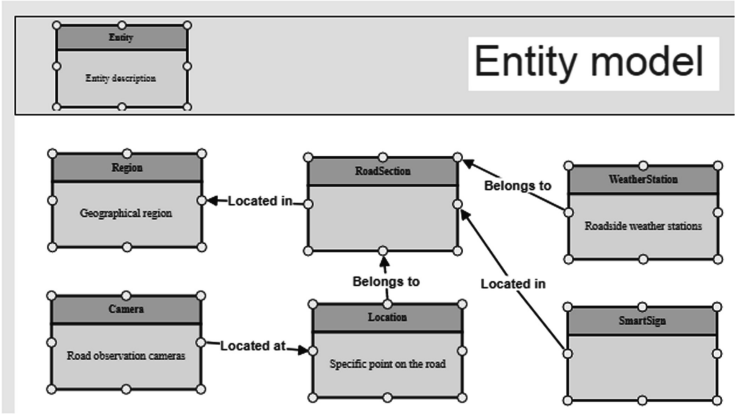

Fig. 4. The entity model as specified in the BaSeCaaS platform

The entities defining the WRM problem are specified in the Entity model (Fig. 4). The road maintenance is performed for specific Road Sections belonging to a Region. There are smart road signs (SmartSigns) providing information for the whole road section as well as weather-stations. The road cameras are installed at specific locations. There is a number of MP. For instance, driving conditions are evaluated according to temperature and precipitation MP provided by road side weather-stations as well as weather service. However, the weather service provides data only at the regional level. Aggregation of measurements is performed according to the relations specified in the entity model.

The BaSeCaaS platform is domain-independent and can be used various use cases. Management of distributed data centers and identification of security threats also has been analyzed.

4 Technical Solution

The technical solution underlying BaSeCaaS consists of three layers (Fig. 5). The platform is model-driven and both information demand and data supply are specified in a form of data models. MP and CE characterize certain entities in the problem domain. These entities and their relationships are defined in the entity model (Fig. 4). MP and CE are also specified and there is a number of predefined filtering, sliding and aggregation functions. Every entity has several instances and entities are used to define dimensions of MP and CE while instances are used as dimensions' indices. For example, entity is a weather-station and MP is temperature, then a single data stream contains temperature values from multiple weather-stations (i.e., instances of the entity). If several data streams are used to compute values of CE or Adjustment trigger then composition models are used to specify the way data streams are joined together. The joint is made along the matching dimensions.

The models are used to configure services. The Orchestration service controls stream processing workflow and ensures delivery of data streams from publishers to subscribers. Messaging topics are created according to the model as well as subscribers and publishers. MP archiving service stores measurements for batch analysis. It is configured according to the archiving specification in the MP model. Adjustment locking service controls frequency of adjustment calls (e.g., adjustment can be invoked for the next time only after a specific time period).

The streaming is implemented using the Spark framework (https://spark.apache.org). A separate Spark job is created for every MP and CE according to the data stream processing model. In the case of high, workload these jobs are horizontally scaled in the cloud computing environment. Adjustments are evaluated using the Computing cluster implemented using Docker containers (https://www.docker.com). The containers are also setup according to the data stream processing model. Containerization allows for flexible choice of adjustment implementation technologies and scalability of intensive computations. The Queuing service is implemented using Kafka (https://kafka.apache.org) supporting scalable processing of high volume data streams. Casandra (http://cassandra.apache.org/) database is used for persistent data storage of archived MP.

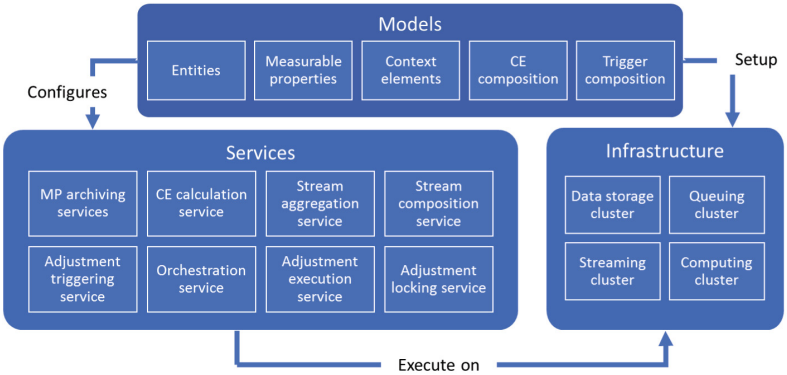

Fig. 5. Layers of the BaSeCaaS platform

5 Demonstration

The WRM case is implemented as demonstration following the design process as showed in Fig. 2. Figure 6a shows the winter road maintenance CE created in the system as well as dialogs for specifying context element composition and calculation. A Spark job is created for every context element. For instance, the context element DrivingConditionsCE is created to characterize current perception of driving condition ranging from normal to poor. This context element is mainly used to provide warnings to various stakeholders. These warnings are implemented as adjustments (Fig. 6b). RoadConditionsWarrningAdj sends a message to the data driven application that context has changed. The triggering rule depends on value of the context elements as specified in the trigger composition. The adjustments can be frozen for a specific period to avoid excessive messaging. The adjustment can be implemented using various technologies and JavaScript is used in this case. The event log keeps trace of adjustments invoked. It is important to note that different data driven applications might need different responses to changes in the context and the platform is able to provide specific adjustments. The adjustments are not limited notifications and complex adaptive logics can be specified within their containers. In the road maintenance case, one of the adjustments evaluates a need to deice the roads.

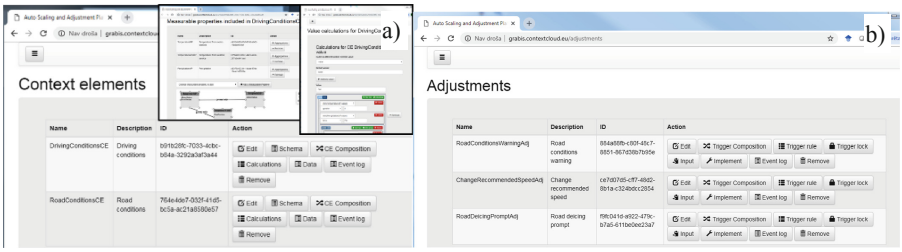

Fig. 6. Definition of (a) context elements and (b) adjustments in the BaSeCaaS platform

Figure 7 shows context element evaluation results for one road section. The Road conditions context element is computed using MP TemperatureMP and PrecipitationMP. One can observe continuous changes of MP while meaningful changes in a sense of varying values of CE occur more rarely. The warning adjustment is triggered only if the locking conditions are met preventing unnecessary nervousness.

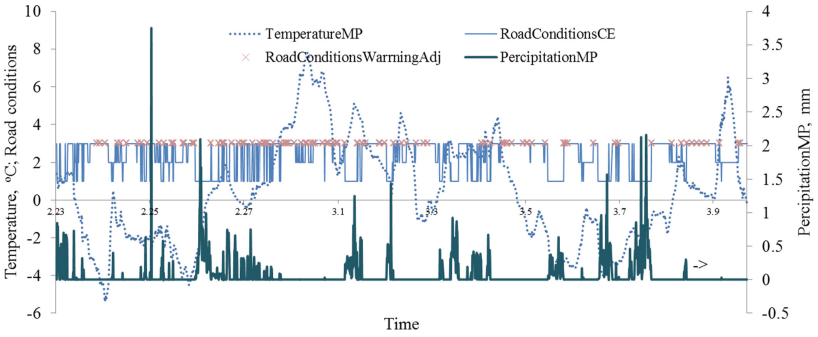

Fig. 7. Sample MP measurements and CE evaluation results for a selected road section

6 Conclusion

BaSeCaaS simplifies development of data streams processing solutions. It supports model driven specification of key data streams processing functions, configuration of streaming services and automated setup of infrastructure. These are important preconditions for making data streams processing as widely used as classical data processing. CE represent information needs in DDA while MP represent data supply. Various combinations of MP can be used to evaluate CE depending on data availability, privacy, business and other considerations. The adjustments are decoupled from DDA to separate intensive computations and frequently changing adaptation logics from the core application. The platform is also horizontally scalable. Application of the proposed platform has been preliminary also for security monitoring in federated computer networks and providing real-time support to users of enterprise applications. The model driven configuration is restricted to the implemented set of stream processing functions. It is not intended to support all types of stream processing functions out-of-the-box rather branching of the platform for specific application cases or domains is envisioned to support custom requirements. One of the main directions of future development of pre-defined adjustments based on machine learning and data mining.

Acknowledgements. This study was funded in parts by European Regional Development Fund (ERDF), Measure 1.1.1.5 “Support for RTU international cooperation projects in research and innovation”. Project No. 1.1.1.5/18/I/008.

References

1. Philip Chen, C.L., Zhang, C.: Data-intensive applications, challenges, techniques and technologies: a survey on Big Data. Inf. Sci. **275**, 314–347 (2014)
2. L'Heureux, A., Grolinger, K., Elyamany, H.F., Capretz, M.A.M.: Machine learning with big data: challenges and approaches. IEEE Access **5**, 7776–7797 (2017)
3. Sandkuhl, K., Stirna, J.: Capability Management in Digital Enterprises. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-90424-5
4. Cugola, G., Margara, A.: Processing flows of information: from data stream to complex event processing. ACM Comput. Surv. **44**, 3 (2012)
5. Gorawski, M., Gorawska, A., Pasterak, K.: A survey of data stream processing tools. In: Czachórski, T., Gelenbe, E., Lent, R. (eds.) Information Sciences and Systems 2014, pp. 295–303. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-09465-6_31
6. Esposito, C., Ficco, M., Palmieri, F., Castiglione, A.: A knowledge-based platform for big data analytics based on publish/subscribe services and stream processing. Knowl.-Based Syst. **79**, 3–17 (2015)
7. Auer, S., et al.: The BigDataEurope platform – supporting the variety dimension of big data. In: Cabot, J., De Virgilio, R., Torlone, R. (eds.) ICWE 2017. LNCS, vol. 10360, pp. 41–59. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-60131-1_3
8. Dey, K.C., Mishra, A., Chowdhury, M.: Potential of intelligent transportation systems in mitigating adverse weather impacts on road mobility: a review. IEEE Trans. Intell. Transp. Syst. **16**, 1107–1119 (2015)