Towards a Methodology for Evaluating Big Data Platforms

Evangelia Kavakli, Rizos Sakellariou
School of Computer Science
University of Manchester, UK

Vlado Stankovski
Faculty of Civil and Geodetic Engineering
University of Ljubljana, Slovenia

***Abstract*—In recent years, several new multipurpose Big Data platforms have emerged. They are used in various application domains with diverse requirements. Evaluating complex Big Data solutions is not a trivial task, due to the need to assess their utility in both quantitative and qualitative terms based on existing use cases. In this short paper, we discuss the requirements and the methodology for such an evaluation. We also discuss how benchmarking could be part of such an evaluation methodology.**

I. Introduction

Several new Big Data platforms that provide, as-a-service, analytic capabilities for different industries and use cases have emerged or are being developed. Such Big Data platforms provide end-to-end business solutions and are often built with specific use cases in mind. All such platforms have to be evaluated regarding their suitability from various viewpoints. Traditionally, a key aspect of this evaluation relies on assessing performance through Big Data benchmarking [1], [2], where specific workloads or datasets of interest may be used to measure various performance-related aspects. Benchmarks may assess reasonably well the capabilities of a Big Data platform with respect to some or all the so-called V characteristics of Big Data (e.g., volume, velocity, variety, veracity, etc). However, benchmarks may be of little help when trying to assess how the requirements of specific use cases are met by a Big Data platform (which has been designed to cater for these use cases). This assessment may require the consideration of a range of qualitative and quantitative aspects, which (in addition to performance) may include usability (e.g., effectiveness, user satisfaction, ease of replication, etc), compliance with regulatory frameworks (e.g., data may have to stay within certain administrative domains, anonymization, etc), reproducibility of information, etc. In turn, this may require a holistic evaluation strategy that can assess a Big Data platform from both a qualitative and a quantitative perspective. This strategy can take into consideration the requirements of the various use cases, the properties of the Big Data platform with respect to the use cases, the suitability of different Big Data Benchmarks, as well as best practice software testing guidelines. In brief, an evaluation methodology should aim to capture Non-Functional, Quality of Service and Quality of Experience requirements to fully evaluate a Big Data platform.

This short paper, representing work-in-progress, is trying to list the key principles and the main phases that should drive the development of such an evaluation methodology. As Big Data platforms become more pervasive, yet they are getting developed with specific industries and use cases in sight, it is getting important to look actively beyond benchmarking into holistic Big Data platform evaluation methodologies. These can enable quick and successful end-users' and stakeholders' acceptance of any new Big Data platform designed.

II. Key principles

The proposed evaluation methodology works in phases, where one moves from the evaluation of the characteristics of the overall platform (using both quantitative and qualitative variables and benchmarks) towards the evaluation of particular deployments in specific real-world settings. In both phases, identifying the evaluation variables to use is informed by capturing the concerns of different stakeholders and analyzing their use cases. This can be done through mixed-mode surveys that may combine structured interviews, questionnaires and so on. When analysing use cases it is necessary to get insight into important aspects related to evaluation, such as:

1) The operational context of the Big Data applications including identification of relationships and dependencies with other systems and the environment.
2) Information flows, particularly data formalisms and models, data flow, and how data is stored and handled.
3) Lists of Big Data platform functionalities, which represent the main aspects of the architecture, the interfaces and the interactions among the architectural components that should be validated.
4) Lists of Non-Functional requirements, Quality of Service and Quality of Experience requirements that must be validated in order to prove utility of the developed architecture in real world practice.
5) The actual deployment in specific industrial settings, which must take into account the ground reality, including, for example, the actual types, quantity, set-up and arrangement of sensors, data sources, computing systems, databases, networking dependencies and similar.

III. Evaluation Methodology

Fig. 1 illustrates the proposed methodology that aims at evaluating and validating: (i) a Big Data platform, and (ii) its implementation with respect to specific use cases. The basic steps of the methodology are described next.

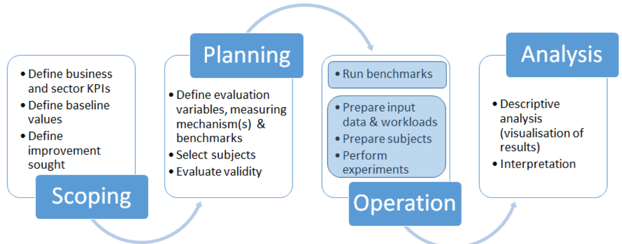

Fig. 1. Evaluation methodology

Scoping is the first step, where we scope the evaluation in terms of business Key Performance Indicators (KPIs) for assessing the achievement of use case objectives. Such KPIs may have been defined in a user requirements elicitation phase. Example KPIs include data movement times, end-to-end analytics time and similar. In addition, for each KPI the baseline (current) value and improvement sought is also defined at this stage. The scoping step is designed, so that it is able to calculate the KPIs based on appropriate measurements.

Planning is the phase where the design of suitable experiments is determined, in terms of the instrumentation (variables to be measured and associated measuring mechanisms) and evaluation subjects that will perform the measurement. Evaluation variables may be relevant to one, more or all of the platform functionalities. Any issues affecting the outcome of the evaluation are to be acknowledged and considerations should be made to ensure the validity of measurements.

Operation of the evaluation follows from the design. In this step, measurements are recorded, analysed and interpreted (with respect to the improvements sought in the scoping step) in the final *Analysis* step. Most specifically, operation considers two phases. The first phase aims at a quantitative validation of the Big Data platform in part and as a whole by using specific benchmarks. It proceeds as follows. For each of the identified use cases specific workloads are selected. The workload types, such as graph analytics, artificial intelligence, data warehouse, streaming, etc., should be well documented and freely available as part of existing benchmarks. Then, the evaluation continues with a gradual increase of the complexity through: (a) definition and experimentation with micro benchmarks; (b) combinations of workloads which are more representative of the specific use cases; and (c) complex end-to-end benchmarking, which involves high-level and business aspects of the specific use cases. The last phase mainly aims at a qualitative evaluation of a Big Data platform in the business context and against the identified business requirements.

A key implication of the last part of the methodology is that benchmarking is not viewed simply as a process of measuring some metrics against specific workloads and/or datasets. Instead, it is an evolving process where several levels and viewpoints are considered to evaluate a platform in relation to its characteristics, use cases, and the expectations of the stakeholders. From the various initiatives concerning Big Data benchmarking, BigDataBench 4.0 [3] fits well into this vision. The reason is that BigDataBench is based on the notion of *dwarfs*, which are abstractions of frequently appearing units of computation, and help achieve scalability of the evaluation. Eight so-called dwarfs exist in BigDataBench 4.0.

The specification of the benchmarks is separate from the implementation. Each use case should be modelled independently from the underlying Big Data system and benchmarks (or dwarfs) should be carefully selected to fit in style and scope what is being used by the Big Data platform's users. Following this, combinations of such dwarfs are formed and may be regarded as component-level benchmarks. These can be understood as representative workloads in different application domains. This process may eventually culminate with the analysis of more complex end-to-end applications. This will cater for additional aspects, such as organisational aspects related to the deployment of the Big Data platform. The best way to view this process is from the top of a pyramid, where individual dwarfs are tested against the individual use cases, as shown in Fig. 2.

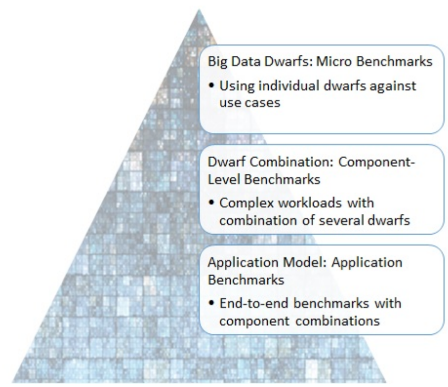

Fig. 2. Benchmarking process view

IV. Summary

Following the elicitation of use case requirements, which can guide the specification of a Big Data platform, this short paper has focused on appropriate processes, variables and mechanisms for evaluating a given platform and its suitability to diverse use cases. The proposed methodology is based on information that is collected from actual users of the Big Data platform as well as different technology providers that may integrate their solutions into the Big Data platform. Benchmarking itself is envisaged as an evolving process, which starts from small individual micro-benchmarks and gradually extends to full scale end-to-end evaluation scenarios. The BigDataBench 4.0 benchmark suite has been described as a potential candidate to be used for quantitative evaluation.

References

[1] T. Ivanov, T. Rabl, M. Poess, A. Queralt, J. Poelman, N. Poggi, and J. Buell, "Big data benchmark compendium," in *Performance Evaluation and Benchmarking: Traditional to Big Data to Internet of Things*. LNCS vol. 9508, Springer, 2016, pp. 135–155.

[2] R. Han, L. K. John, and J. Zhan, "Benchmarking big data systems: A review," *IEEE Transactions on Services Computing*, pp. 580–597, 2018.

[3] "BigDataBench," http://prof.ict.ac.cn/BigDataBench.