An Approach to Benchmarking Industrial Big Data Applications

Umeshwar Dayal(✉), Chetan Gupta, Ravigopal Vennelakanti, Marcos R. Vieira, and Song Wang

Big Data Research Lab, Hitachi America, Ltd., R&D, Santa Clara, CA, USA
umeshwar.dayal@hal.hitachi.com,
{chetan.gupta,ravigopal.vennelakanti,
marcos.vieira,song.wang}@hds.com

Abstract. Through the increasing use of interconnected sensors, instrumentation, and smart machines, and the proliferation of social media and other open data, industrial operations and physical systems are generating ever increasing volumes of data of many different types. At the same time, advances in computing, storage, communication, and big data technologies are making it possible to collect, store, process, analyze and visualize enormous volumes of data at scale and at speed. The convergence of Operations Technology (OT) and Information Technology (IT), powered by innovative data analytics, holds the promise of using insights derived from these rich types of data to better manage our systems, resources, environment, health, social infrastructure, and industrial operations. Opportunities to apply innovative analytics abound in many industries (e.g., manufacturing, power distribution, oil and gas exploration and production, telecommunication, healthcare, agriculture, mining) and similarly in government (e.g., homeland security, smart cities, public transportation, accountable care). In developing several such applications over the years, we have come to realize that existing benchmarks for decision support, streaming data, event processing, or distributed processing are not adequate for industrial big data applications. One primary reason being that these benchmarks individually address narrow range of data and analytics processing needs of industrial big data applications. In this paper, we outline an approach we are taking to defining a benchmark that is motivated by typical industrial operations scenarios. We describe the main issues we are considering for the benchmark, including the typical data and processing requirements; representative queries and analytics operations over streaming and stored, structured and unstructured data; and the proposed simulator data architecture.

1 Introduction

Today, we are at the dawn of transformative changes across industries, from agriculture to manufacturing, from mining to energy production, from healthcare to transportation. These transformations hold the promise of making our economic production more efficient, cost effective, and, most importantly,

T. Rabl et al. (Eds.): WBDB 2014, LNCS 8991, pp. 45–60, 2015.
DOI: 10.1007/978-3-319-20233-4_6

sustainable. These transformations are being driven by the convergence of the global industrial system (Operations Technology (OT)) with the power of integrating advanced computing, analytics, low-cost sensing and new levels of connectivity (Information Technology (IT)).

Through the increasing use of interconnected sensors and smart machines and the proliferation of social media and other open data, industrial operations and physical systems produce a very large volume of continuous stream of sensor, event and contextual data. This unprecedented amount of rich data needs to be stored, managed, analyzed and acted upon for sustainable operations of these systems. Big data technologies, driven by innovative analytics, are the key to creating novel solutions for these systems that achieve better outcomes at lower cost, substantial savings in fuel and energy, and better performing and longer-lived physical assets.

Opportunities to create big data solutions abound in many industries (e.g., power distribution, oil and gas exploration and production, telecommunication, healthcare, agriculture, mining) and in the public sector (e.g., homeland security, smart cities, public transportation, population health management). To realize operational efficiencies and to create new revenue-generating lines of business from the deluge of data requires the convergence of IT $\times$ OT. The convergence of these two technologies can be obtained by leveraging an analytics framework to translate data-driven insights from a multitude of sources into actionable insights delivered at the speed of the business. Thus, innovations in analytics will be required: (1) to deal with the vast volumes, variety, and velocity of data; and (2) to create increasing value by moving from descriptive or historical analytics (e.g., what has happened and why?) to predictive analytics (e.g., what is likely to happen and when?) and finally to prescriptive analytics (e.g., what is best course of action to take next?).

In this paper, we provide a detailed discussion of a proposed benchmark for industrial big data applications. Existing benchmark proposals either focus on OLTP/OLAP workloads for database systems or focus on enterprise big data systems. Our objective is to define a benchmark for IT $\times$ OT big data applications in industrial systems. We recognize that proposing such a benchmark is a complex and evolving task. To the best of our knowledge, this paper is the first to outline a systematic approach to defining a benchmark for industrial big data applications.

The paper is organized as follows. Section 2 details existing benchmarking proposals for various application domains. Section 3 describes the characteristics of IT $\times$ OT Big Data Applications. Section 4 provides an overview of the main features of our proposed benchmark. Section 5 details the initial benchmark architecture implementation. Finally, Sect. 6 concludes the paper and outlines the future extensions to our proposed benchmark.

2 Related Work

There exist many efforts for developing benchmarks for big data systems, each focusing on evaluating different features. Examples of industry standard

benchmarks are TPC-H [13,18] and TPC-DS [10,14], both developed by the Transaction Processing Performance Council (TPC), the leading benchmarking council for transaction processing and database benchmarks. These two decision support benchmarks are employed to compare SQL-based query processing performance in SQL-on-Hadoop and relational systems [1,6,12,15,17]. Although these benchmarks have been often used for comparing query performance of big data systems, they are basically SQL-based benchmarks and, thus, lack the new characteristics of industrial big data systems.

Several proposals have extended the TPC-H and TPC-DS to deal with new characteristics of big data systems. For example, BigDS benchmark [20] extends TPC-DS for applications in the social marketing and advertisement domains. Han and Lu [8] discuss key requirements in developing benchmarks for big data systems. However, neither of these two proposals defines a query set and data model for the benchmark. HiBench [9] and BigBench [2,4,7] are *end-to-end application-based benchmarks* that extend the TPC-DS model to generate and analyze web logs. Some other works propose *component benchmarks* to analyze specific features or hardware of big data systems (e.g., [12,19]). However, a *component benchmark*, which measures performance of one (or a few) components, has limited scope compared to *end-to-end benchmarks*, which measures full system performance. Additionally, there are proposals of big data tools for benchmarking big data systems (e.g., [16,21]), and even efforts to maintain a top-ranked list of big data systems based on a benchmark [3] (similar in concept to the Sort Benchmark competition [11]).

Recently the TPC released the TPC Express Benchmark for Hadoop Systems (TPCx-HS) [5]. This benchmark is based on the TeraSort benchmark, which is a Hadoop-based sort benchmark [11], to measure hardware, operating system and commercial Hadoop File System API.

Common to all of the above benchmarks is that they are designed to measure limited features of big data systems of specific application domains (e.g., decision support, streaming data, event processing, distributed processing). However, none of existing benchmarks covers the range of data and analytics processing characteristic of industrial big data applications. To the best of our knowledge, our proposed benchmark is the first to address typical data and processing requirements, representative queries and analytics operations over streaming and stored, structured and unstructured data of industrial big data applications.

3 IT × OT Big Data Applications

In this section, we first describe the main characteristics of industrial big data systems in terms of: (1) latencies and kinds of decisions; (2) variety of data; and (3) type of operations that occurs in these systems. We then illustrate examples of these characteristics using two industrial applications.

The class of IT × OT big data applications can be understood in terms of the life cycle of data, as illustrated in Fig. 1. In today's architecture, the outermost cycle is the strategy cycle, where the senior executives and back-office staff use

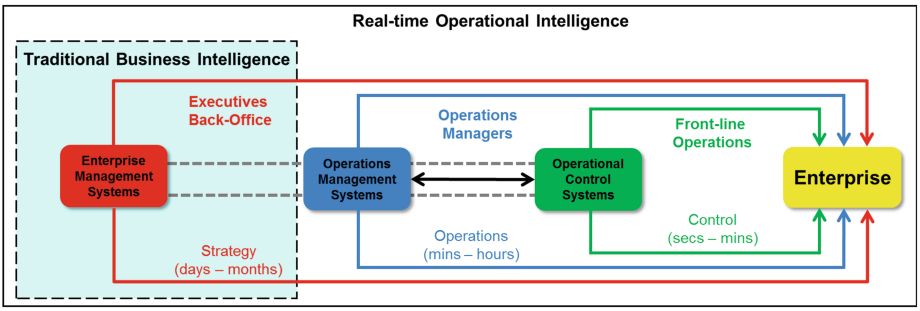

Fig. 1. For IT×OT applications "Operational Intelligence" is the key requirement. When decreasing time scales for responses (decision latencies), we have an increase in automation for big data systems.

historical data stored in an Enterprise Data Warehouse (EDW) to take long term decisions (i.e., very long latency – days or months). In this class of applications the data is stored in an EDW which has been already processed, normalized and structured using ETL tools. Furthermore, the operations available in these systems are limited to historical data analysis.

We often see application needs moving from traditional business intelligence to a more real-time operational intelligence. Thus, we want to be able to make decisions in order of minutes or hours for **operations management systems**, and if we want to decrease decision latencies, in the order of seconds or minutes, we have **operational control systems**. Along with, decreasing the latencies and kinds of decisions when moving to a more real-time operational intelligence, we also observe an increase in variety of data (e.g., stored/streaming data, structured/semi-structured/unstructured data) and in the complexity of operations (e.g., ranging from advanced queries to time series analysis to event correlation and prediction) in these systems. These advanced features in real-time operational intelligence are illustrated in the middle and innermost cycles in Fig. 1. In the middle cycle, or daily operations cycle, the operations managers take day-to-day decisions, such as inventory management, whereas in the innermost cycle, the control or the operations management cycle, which is aimed at responding in real time to changes in the state of the enterprise for efficient management of resources.

The traditional business intelligence approach to building applications has the main drawback that EDWs only store part of the data, while optimal decision making at any time scale is dependent on all the data available to the enterprise. Hence, the new architecture for IT × OT application can exploit the emerging big data architectures to make available all the data collected from the enterprise and perform analytics over this data for decision making. In other words, we are moving from traditional business intelligence towards real-time operational intelligence, where a single system is able to provide decision making at any time scale. We illustrate this with examples from two industries. For both of

these industries we illustrate the data and problem requirement that cannot be address by today's enterprise architectures.

3.1 Electric Power Industry Application

Applications in the electric power industry are collecting data at fine granularities, from a large number of smart meters, power plants, and transmission and distribution networks. The variety of data collected also ranges from streaming data (e.g., data from a large volume of smart meters and sensors, energy trading time series) to stored data (e.g., daily billing information). Moreover, applications in this domain are increasing in complexity, from simple historical queries to more advanced analytics on streaming data (e.g., list the neighborhoods with the highest increase energy demands compared to the same period last week), to time series and event predictions (e.g., when and where a power failure will occur?).

In terms of a system for operational intelligence, in the outermost cycle (illustrated in Fig. 2) on a long term basis we are concerned with problems such as matching power supply with demand for a region, allocating new suppliers, and managing future derivative contracts. In the daily cycle, we are concerned with daily billing and managing daily supply with demand. In the operation management cycle, we address problems such as detection of outages, dynamic pricing, consumption optimization, personalized electric power usage recommendations, among others.

Today, in the electricity industry, there are different systems for each of these granularities of decision making. Furthermore, the raw smart meter data is stored only at aggregate level, and the original raw data is discarded, consequently not available for decision making.

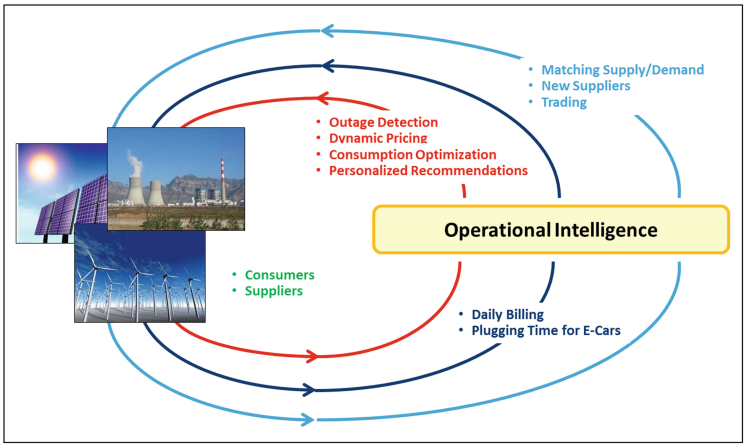

Fig. 2. Smart power grid application scenario.

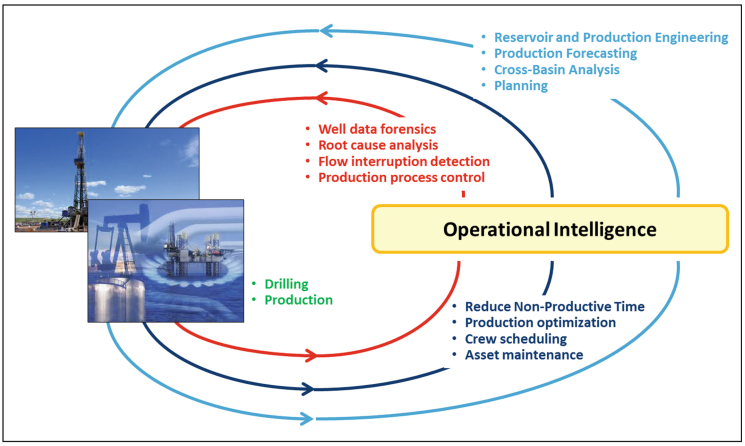

Fig. 3. Smart oil and gas fields application scenario.

3.2 Oil and Gas Fields Application

Applications in the oil and gas industry are collecting data at fine granularities, from electric submersible pumps, oil and gas pipelines, sensors in well reservoirs, drilling rigs, oil platforms, and oil and gas refineries. The variety of data being collected is also changing from only historical to streaming data (e.g., data from multiple sensors installed in a network of oil rigs), and from only structured to also include unstructured data (e.g., analysis on data from scientific papers, operator notes, technical reports). Also, the complexity on the type data operations are increasing, from queries on historical data to more data-driven intensive computation (e.g., find the oil and gas plays with high hydrocarbon content, find the best completion technique for a given unconventional well).

In terms of the outermost cycle (as shown in Fig. 3) on a long term basis we are concerned, for example, with problems such as management of reservoir and production, oil and gas production forecasting, analysis of different basins, and production planning. In the daily cycle, we are concerned with reducing the non-productive time, optimizing the production, crew scheduling, and asset maintenance. In the operation management cycle, we address problems such as well as data forensics, root-cause analysis, flow interruption detection in pipelines, production process control, among others.

We saw with the examples from electric power and oil and gas industries the common characteristics industrial applications share. To handle the three characteristics described above, efficient, scalable and different big data architectures are needed. In order to test these complex and different architectures for industrial applications we need a generic approach to benchmarking these applications.

4 Features of Industrial Big Data Benchmark

We now present the required features needed to build an industrial big data benchmark. We first detail an **application scenario** to illustrate the different requirements of the IT × OT applications. We then abstract the application scenario and develop an **activity model**, which gives a definition of the important features and their relationships in the application scenario. We then discuss the **streaming and historical data** that is potentially collected for such applications, and the **queries and analytics operations** that can be performed over the data. Finally, we discuss the **evaluation measures** that should be used to evaluate the benchmark and the **simulator parameters** that need to be set for the simulator and data generator in order to obtain the benchmark with the desired properties.

4.1 Application Scenario

We select an application scenario from the surface mining industry because it is representative of many industrial IT × OT systems: it is characterized by a set of field operations involving physical objects, such as stationary and moving equipment; the operational activities generate data of diverse types, and require a variety of analytics processing to provide operational intelligence at different latencies, both in the field and at remote operations centers.

Surface mining is a category of mining in which soil overlying valuable natural resources is removed. One process commonly employed in the surface mining industry is known as *open-pit mining*. In open-pit mining, rocks are first blasted where the raw material is to be found. Once the blast has taken place, the regular mine operation happens in shifts. During a typical shift several shovels or loaders pick up either the raw material or the waste from the blast site and load it into trucks or haulers. If the truck is hauling raw material it proceeds to the processing plant or the stockpile; if the truck is hauling waste it proceeds to a dump site. Once the truck is emptied, it goes back to the loading unit (shovel or loader) and the cycle (referred to as an operation cycle) starts again.

At any point in the mine, there are several shovels and a much larger number of trucks operating. In a modern mine, data is constantly being collected from various different equipment. From a mine managers' perspective, there are several primary objectives that need to be optimized during the operation of the mine: (1) getting the correct mix of different grades of the raw material based on the market conditions; (2) making sure the equipment is available in the field to perform the activities of operation processes; and (3) making the most efficient use of the equipment that is available in the mine.

An abstract view of data collection and processing for such field application scenario is illustrated in Fig. 4. Data from various different field equipment is constantly relayed via field sensors to a Remote Operations Center (ROC). Various decisions support applications are built on top of the collected data.

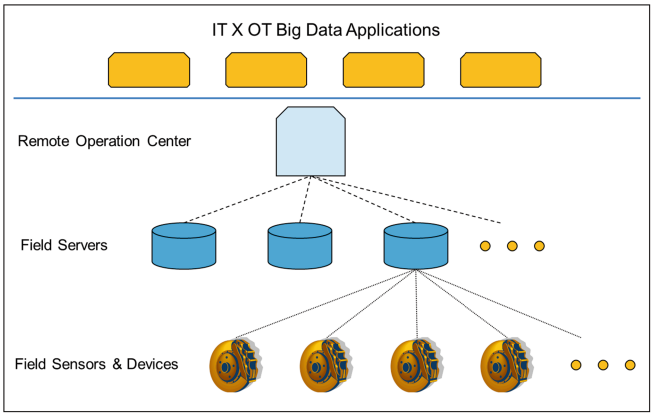

Fig. 4. Generic field IT × OT big data application scenario.

4.2 Activity Model

From the described application scenario we can highlight some of the high level requirements for an IT × OT application:

1. Data ingestion and storage capabilities from several different processes and equipment;
2. Data processing features from different data types and sources;
3. Encoding some domain related knowledge regarding the operations and activity on the field site;
4. Data analysis from simple queries to advanced analytics with different execution and response types;
5. Some simple to advanced visualization features from the data operations and activities[1].

From the above list of requirements, we further abstract and define an activity model:

- **Operational process** is a Directed Acyclic Graph (DAG) of activities, where manufacturing operations are the vertices and the edges represent a process sequence. In the mining use case, an operational process for loaders and haulers, see Fig. 5, has the following graph representation: *loading* → *hauling full* → *queuing at dump* → *dumping* → *hauling empty* → *queuing at loader*;
- Each **activity** is performed by **equipment**, which acts on an **object**. In the mining use case, **equipment** is a shovel or a hauler, and **object** is waster or raw materials;
- **Equipment** might be in **stationary** or **moving** states. Shovels are often in **stationary** state and haulers are in **moving** state;
- **Sensors** measure attributes of **equipment** and attributes of **object**;

[1] Visualization features are not covered in this version of our proposed benchmark.

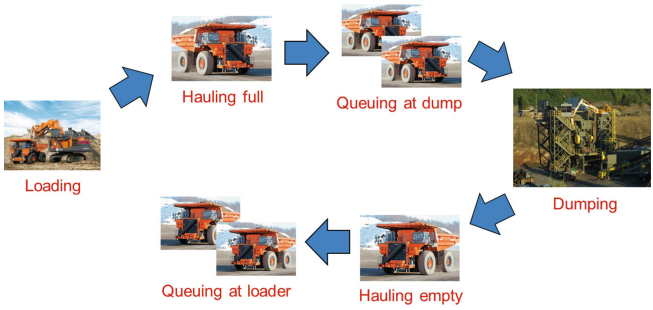

Fig. 5. Operational process of a haul cycle represented as a DAG of activities.

- **Events** signal start and finish of **activities**;
- **Equipment cycle** is one complete sequence of **activity** for **equipment**;
- **Operational cycle** is one complete sequence of equipment **activities** for an **operational process**;
- **Work shift** consists of several different **operational cycles** running concurrently and repeatedly.

Based on the activity model, we further detail other characteristics of the proposed benchmark for IT × OT applications.

4.3 Streaming and Historical Data

There are several different types of data that are generated by OT systems. Our proposed benchmark uses data generated by a simulator that simulates the operation processes (described in Sect. 5). Historical data is stored in the data management and processing layers (e.g., RDBMS, Key-Value Store, Hadoop) and real time data is streamed. In the following we describe the data schema, shown in Fig. 6, for the historical and streaming data:

1. Operational data from equipment includes both real-time stream data (e.g., from current work shift) and historical, stored data (e.g., from n previous work shifts):
   - Event data generated from equipment (begin/end activities, other operational events):
     ActivityEventMeasure (equipmentId, eventId, beginTimestamp, endTimestamp)
   - Measurements from sensors on equipment:
     EquipmentSensorMeasure (sensorId, value, timestamp)
   - GPS Sensors on moving equipment:
     MovingSensorLocationMeasure (sensorId, GPSCoordinates, timestamp)
   - Field sensors (e.g., beacons):
     FieldEquipmentMeasure (sensorId, equipmentId, timestamp)

   - Equipment data:
     EquipmentFleetMeasure (equipmentId, fleetId, timestamp)
2. Non-equipment operational data:
   - Measurements from sensors that measure attributes of object being acted upon:
     ObjectAttributeEstimationMeasure (sensorId, value, timestamp)
3. Object operational data:
   - Production target within a shift:
     ShiftProdutionTargetMeasure (shiftId, targetProduction)
   - Relationship between activity and event type:
     EventActivityMapping (acitivityId, eventId)
4. External operational data:
   - Ambient sensor measures:
     AmbientSensorMeasure (sensorId, value, timestamp)
5. Process operational data:
   - Shift start and end times:
     Shift (shiftId, beginTimestamp, endTimestamp)
6. Operator data:
   - Operational notes:
     OperationalNotes (authorId, timestamp, noteValue)
7. Business data:
   - Attributes of equipment:
     Equipment (equipmentId, type, make, year, properties)
   - Sensor attributes:
     Sensor (sensorId, measurementType, make, model, properties)
     EquipmentSensorRelationship (sensorId, equipmentId, timestamp)
     FieldSensorLocation (sensorId, GPSLocation, timestamp)
   - Relationship between different equipment:
     EquipmentAdjacency (equipmentId1, equipmentId2, activityId)
   - Process model:
     ActivityGraph (activityId1, activityId2, adjacencyRelationship)
8. There may exist other data sources:
   - External business data (e.g., equipment cost):
     CostOfEquipment (equipmentId, cost, timestamp)
   - Manufacturer's manual:
     EquipmentUserManual (equipmentId, partId, description)

4.4 Queries and Analytics Operations

Having defined the data schema for IT × OT benchmark, we now describe the operations on the data, ranging from simple SQL-based queries to more complex analytics operations. Even for a single type of operation (e.g., a single SQL-based query), the possible range of queries and analytics operations form a large space. To systematically study this, we describe the queries and analytics operations in terms of the following properties:

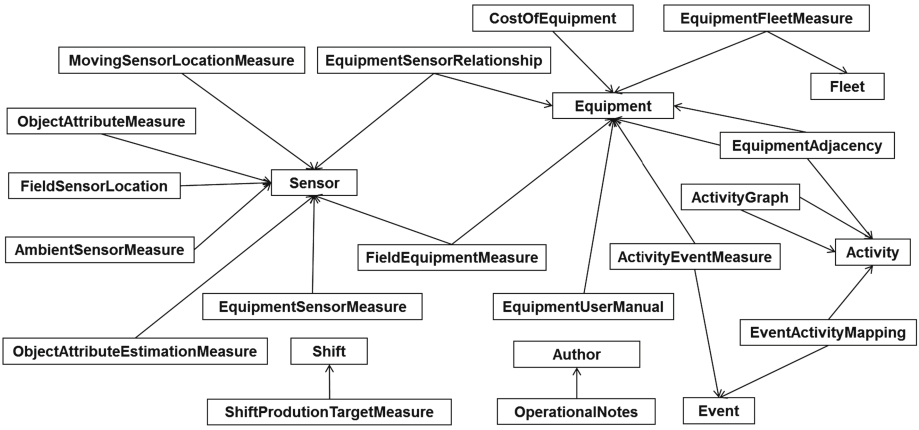

Fig. 6. Logical data schema.

1. Data types:
   - Sensor data
   - Event data
   - Log data
   - Geospatial data
   - Business data
2. Data states:
   - Data in motion
   - Data at rest
   - Hybrid data state
3. Query execution types:
   - One-shot query
   - Continuous query
   - Scheduled query
4. Query response types:
   - Real-time query
   - Near real-time query
   - Best effort query

We identified some important types of queries and analytics operations that are executed over the stored and streaming data generated by the simulator. Below we summarize the types of queries and analytics operations:

1. Aggregate functions;
2. Lookup & search operations;
3. ETL operations on new data;
4. OLAP on stored data;
5. Data mining and machine learning operations:
   - Outlier detection;

- Time series prediction;
- Event prediction;
- Key Performance Indicator (KPI) prediction;
- Spatio-temporal pattern detection;
- Diagnostics and root cause analysis.

6. Planning and optimization operations.

We now present several sample benchmark queries and analytics operations. These sample operations are defined in terms of the query description and primary data entity accessed to answer such queries:

1. Aggregate function queries performed over time: last $t = 5$ min (sliding window move by 1 min), current shift (landmark window), or previous shift (tumbling window), compute:
   (a) Total number of activities by each equipment:
   (ActivityEventMeasure, EventActivityMapping)
   (b) Total number of operation cycles:
   (ActivityEventMeasure, ActivityAdjacency, EventActivityMapping, ActivityGraph)
   (c) Total number of equipment cycles per equipment per equipment type:
   (ActivityEventMeasure, Equipment, EventActivityMapping, ActivityGraph)
   (d) Total distance moved per each moving equipment per operation cycle:
   (ActivityEventMeasure, MovingSensorLocationMeasure, EquipmentSensorRelationship)
   (e) Top-k average high measure from equipment:
   (EquipmentSensorMeasure, EquipmentSensorRelationship)
2. Detection of threshold events:
   (a) Alert if the object attribute measure is outside the bounds (mean +/- 2σ over a fixed window of last shift):
   (ObjectAttributeEstimationMeasure)
   (b) Alert if wait time for any activity is above its expected value bounds (mean +/- 2σ over a sliding window of last k hour sliding by 1 h):
   (ActivityEventMeasure)

3 Detection of outlier events:
   (a) Real-time outlier detection for machines based on equipment sensor health data:
   (EquipmentSensorMeasure, EquipmentSensorRelationship)

4. Detection of general events:
   (a) Alert if there is an order violation in terms of activity order:
   (ActivityEventMeasure, ActivityGraph, EventActivityMapping)
   (b) Alert if moving equipment visits a specified field sensor less than once during an operational cycle:
   (FieldEquipmentMeasure, ActivityEventMeasure)
5. Production estimation query:
   (a) Predict if an object attribute measure will miss the shift target:
   (ObjectAttributeEstimationMeasure, ShiftProductionTargetMeasure)

6. Facility optimization query:
   (a) Maximize the utilization of the most expensive equipment in the working shift:
   (Equipment, CostOfEquipment, ActivityData)
7. Sensor health query:
   (a) Detect faulty sensors by comparing to similar sensors:
   (EquipmentSensorMeasure, EquipmentSensorRelationship, Sensor)
8. Equipment event prediction:
   (a) Predict an equipment sensor health event with a time horizon of t hours:
   (EquipmentSensorMeasure, ActivityEventMeasure, EquipmentAttribute, EquipmentSensorRelationship, AmbientSensorMeasure)
9. Recommended action operation:
   (a) For a predicted equipment maintenance event (e.g., downtime), recommend best action:
   (OperatorNotes, EquipmentSensorMeasure, ActivityEventMeasure, EquipmentAttribute, EquipmentSensorRelationship, AmbientSensorMeasure)

4.5 Evaluation Measures

Below we propose evaluation measures for each query and analytics operations previous described:

1. For aggregate functions over time windows (query type 1):
   - **Response time**: report the result within t time units of close of window;
   - **Accuracy**: results should be correct for every query.
2. For threshold, outlier and general event detection (queries type 2–4):
   - **Response time**: report the event within t time units of event occurring;
   - **Accuracy**:
     - All event occurrences should be detected;
     - Report f-measure for outlier events.
3. For production estimation query (query type 5):
   - **Response time**: report the production estimation within t time units of work shift;
   - **Accuracy**: report f-measure for estimation.
4. For facility optimization query (query type 6):
   - **Response time**: report the utilization value within t time units of work shift;
   - **Relative error**: report the difference between the reference and reported maximum utilization value.
5. For sensor health query (query type 7):
   - **Response time**: report the sensor health within t time units of close of window;
   - **Accuracy**: results should be correct for every query.

6. For equipment event prediction (query type 8):
   - **Response time**: report the prediction within t time units of close of window;
   - **Accuracy**: report f-measure for prediction.
7. For recommended action operation (query type 9):
   - **Response time**: report the utilization value within t time units of work shift;
   - **Accuracy**: report f-measure for recommended action.

4.6 Simulation Parameters

As part of the proposed benchmark we provide a simulator and data generator. The activity simulator will simulate the operational processes (e.g., haul cycles) and from that generate the relevant data (e.g., sensor, event, log data). The simulator and data generator is capable of producing large amounts of data in a scalable and high performance fashion. The input parameters are described below:

1. Number of static and moving equipment in the field;
2. Number of sensors for each equipment;
3. Number of field sensors;
4. Sampling frequency of each sensor;
5. Event rates of non-activity events;
6. Number of activities in an equipment cycle;
7. Number of activities in an operational cycle;
8. Number of operational cycles in a shift;
9. Number of shifts in the entire benchmark;
10. Volume of historical and business data.

5 Benchmark Implementation

In our proposed benchmark architecture, shown in Fig. 7, we provide two modules as part of the benchmark: the simulator and data generator and the evaluation modules. The simulator and data generator module receive a set of input

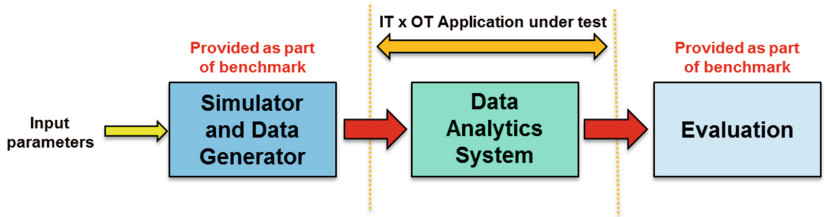

Fig. 7. Benchmarking IT × OT big data applications: simulator and data generator and evaluation modules are provided as part of the benchmark.

parameters and output a set of (streaming/historical) logs, events, business data. The set of output data includes different data types, from static business data to real-time stream and event data. This set of data is provided as input to the IT × OT application under test. The second module evaluates and generates reports on the output data produced by the IT × OT application.

6 Conclusion and Future Works

In this paper, we present the first steps towards the definition of a benchmark for industrial big data applications. Previous benchmarks do not cover the wide range of data, queries and analytics processing characteristics of industrial big data applications. Our proposed benchmark is motivated by a typical industrial operations scenario, including the typical data types and processing requirements, representative queries and analytics operations, and a data generator and simulator architecture.

As a next step in this research we are performing the following tasks:

1. We have provided a description of a benchmark. Based on the community's feedback we will refine the benchmark and provide precise definition of the set of queries and analytical operations;
2. We are starting to implement the benchmark. As part of the implementation, we are developing the data generator and simulator modules;
3. We are studying additional benchmarking measures targeting industrial applications (e.g., ease of development, cost, energy efficiency, security, maintainability).

References

1. Abouzied, A., Bajda-Pawlikowski, K., Huang, J., Abadi, D.J., Silberschatz, A.: HadoopDB in action: building real world applications. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 1111–1114 (2010)
2. Baru, C., et al.: Discussion of bigbench: a proposed industry standard performance benchmark for big data. In: Nambiar, R., Poess, M. (eds.) TPCTC 2014. LNCS, pp. 44–63. Springer, Hiedelbeg (2015)
3. Baru, C., Bhandarkar, M., Nambiar, R., Poess, M., Rabl, T.: Big data benchmarking and the BigData top100 list. Big Data J. **1**(1), 60–64 (2013)
4. Chowdhury, B., Rabl, T., Saadatpanah, P., Du, J., Jacobsen, H.A.: A BigBench implementation in the Hadoop ecosystem. In: Rabl, T., Raghunath, N., Poess, M., Bhandarkar, M., Jacobsen, H.A., Baru, C. (eds.) WBDB 2013. LNCS, vol. 8585, pp. 3–18. Springer, Heidelberg (2014)
5. Transaction Processing Performance Council, TPCx-HS, February 2015. www.tpc.org/tpcx-hs/
6. Floratou, A., Minhas, U.F., Özcan, F.: SQL-on-hadoop: full circle back to shared-nothing database architectures. Proc. VLDB Endowment (PVLDB) **7**(12), 295–1306 (2014)

7. Ghazal, A., Rabl, T., Hu, M., Raab, F., Poess, M., Crolotte, A., Jacobsen, H.-A. BigBench: towards an industry standard benchmark for big data analytics. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 1197–1208 (2013)
8. Han, R., Lu, X., Xu, J.: On big data benchmarking. In: Han, R., Lu, X., Xu, J. (eds.) BPOE 2014. LNCS, pp. 3–18. Springer, Heidelberg (2014)
9. Huang, S., Huang, J., Dai, J., Xie, T., Huang, B.: The HiBench benchmark suite: characterization of the MapReduce-based data analysis. In: Proceedings of the IEEE International Conference on Data Engineering (ICDE) Workshop, pp. 41–51 (2010)
10. Nambiar, R.O., Poess, M.: The making of TPC-DS. In: Proceedings of the International Conference on Very Large Data Bases (VLDB), pp. 1049–1058 (2006)
11. Nyberg, C., Shah, M., Govindaraju, N.: Sort Benchmark, February 2015. http://sortbenchmark.org
12. Pavlo, A., Paulson, E., Rasin, A., Abadi, D.J., DeWitt, D.J., Madden, S., Stonebraker, M.: A comparison of approaches to large-scale data analysis. In: Proceedings of the of the ACM SIGMOD International Conference on Management of Data, pp. 165–178 (2009)
13. Poess, M., Floyd, C.: New TPC benchmarks for decision support and web commerce. SIGMOD Rec. **29**(4), 64–71 (2000)
14. Poess, M., Nambiar, R.O., Walrath, D.: Why you should run TPC-DS: a workload analysis. In: Proceedings of the VLDB Endowment (PVLDB), pp. 1138–1149 (2007)
15. Poess, M., Rabl, T., Caufield, B.: TPC-DI: the first industry benchmark for data integration. Proc. VLDB Endowment (PVLDB) **7**(13), 1367–1378 (2014)
16. Rabl, T., Jacobsen, H.-A.: Big data generation. In: Rabl, T., Poess, M., Baru, C., Jacobsen, H.-A. (eds.) WBDB 2012. LNCS, vol. 8163, pp. 20–27. Springer, Heidelberg (2014)
17. Thusoo, A., Sarma, J.S., Jain, N., Shao, N., Chakka, P., Zhang, N., Antony, S., Liu, H., Murthy, R.: Hive - a petabyte scale data warehouse using hadoop. In: Proceedings of the IEEE International Conference on Data Engineering (ICDE), pp. 996–1005 (2010)
18. Transaction Processing Performance Council (TPC), TPC-H benchmark specification (2008). http://www.tpc.org/tpch/
19. Wang, L., Zhan, J., Luo, C., Zhu, Y., Yang, Q., He, Y., Gao, W., Jia, Z., Shi, Y., Zhang, S., Zheng, C., Lu, G., Zhan, K., Li, X., Qiu, B.: BigDataBench: a big data benchmark suite from internet services. In: IEEE International Symposium on High Performance Computer Architecture (HPCA), pp. 488–499, February 2014
20. Zhao, J.-M., Wang, W.-S., Liu, X., Chen, Y.-F.: Big data benchmark - Big DS. In: Rabl, T., Raghunath, N., Poess, M., Bhandarkar, M., Jacobsen, H.-A., Baru, C. (eds.) WBDB 2013. LNCS, vol. 8585, pp. 49–57. Springer, Switzerland (2014)
21. Zhu, Y., Zhan, J., Weng, C., Nambiar, R., Zhang, J., Chen, X., Wang, L.: BigOP: generating comprehensive BigData workloads as a benchmarking framework. In: Bhowmick, S.S., Dyreson, C.E., Jensen, C.S., Lee, M.L., Muliantara, A., Thalheim, B. (eds.) DASFAA 2014, Part II. LNCS, vol. 8422, pp. 483–492. Springer, Heidelberg (2014)