

A MODEL OF BIG DATA ARCHITECTURE ON THE BASE
OF FIWARE COMPONENTS

Alexander Chikalanov*, Leoneed Kirilov^{*,**,#}, Eugenia Kovatcheva*,
Roumen Nikolov*, Elena Shoikova*, Alexander Iliev^{*,***},
Ljubomir Gotsev*

Received on May 25, 2023

Presented by Ch. Roumenin on June 27, 2023

Abstract

The intensive development of hardware and software tools provides many opportunities for solving various real-world tasks, such as working with large data sets. This, in turn, is a powerful incentive and opportunity for the development of different approaches and tools for solving them. After discussing several well-known BigData architectures a model of a BigData architecture is designed. The architecture is organized in three sections: edge section, processing-and-storage section and application section. The model will be implemented using FIWARE components.

Key words: BigData, FIWARE, Internet of Things, smart city, data architectures

1. Introduction. In recent years the research and application area of Big Data has been developing at an increasingly rapid pace. Several factors contribute to this:

- the development of hardware and software tools;

[#]Corresponding author.

This work was supported by grant No BG05M2OP011-1.001-0003, financed by “Science and Education for Smart Growth” operational programme and co-financed by the European Union through the European structural and Investment funds and by the Bulgarian National Science Fund under grant No. KP-06-N52/7.

DOI:10.7546/CRABS.2023.09.10

- the development of powerful algorithms enabling the processing of large data sets;
- mass digitization in modern society.

All this allows nowadays to formulate and solve many new problems requiring work with large data sets. This, in turn, raises a number of questions about their storage, processing, analysis, calculation, visualization, security, etc. [1–5].

BigData can be characterized primarily as a collection of large amount of data. These data can differ in their nature, structure, content, time. The unifying factor is that these data are of interest in aggregate and they are processed in aggregate. BigData is well described in terms of “V’s” [6,7]. The first three V’s are: Volume, Velocity (data generation, data processing), Variety (sources, types, structure, security, completeness). The next V’s are Veracity (data uncertainty), Value (appropriate data and appropriate analysis), Valency (data interconnectivity), Volatility (connected with data retention and first 3 V’s), Validity (data suitability to particular usage).

BigData constantly poses new challenges to researchers and practitioners. One of them is how to organize hardware components and particular software tools in order to handle BigData for different purposes [8,9]. In other words, how the BigData architecture to be organized.

The goal of this paper is to propose a model of new BigData architecture. Further the paper is organized as follows. In the next section we discuss the basic types of BigData architecture with applications. In Section 3 a model of new BigData architecture is formulated and in Section 4 its components are discussed.

2. Basic types BigData architectures. The composition of BigData architecture is closely related with the nature of the organization’s activities now and in the future. It depends on the type and volume of data, data processing and analysis necessary, data usage and storage.

In [8], a generalized model of BigData architecture is presented (see Fig. 1). It consists of the following components:

- Data sources – one or more;
- Data storage – usually distributed file store (data lake);
- Batch processing;
- Real-time message ingestion (for real-time sources) – a simple data store or message ingestion store/buffer for messages;
- Stream processing – process the real-time messages (filter, aggregate etc.) and prepare the data for analysis. Put the stream data into an output sink;
- Analytical data store – prepare data for analysis (structured format) and then apply analytical tools;
- Analysis and reporting – present the results to users. A data modelling layer is possible to be used with appropriate visualization;
- Orchestration – to automate repetitive operations in the format of workflows (source data transformation, data moving, processing, calculation, analysis,

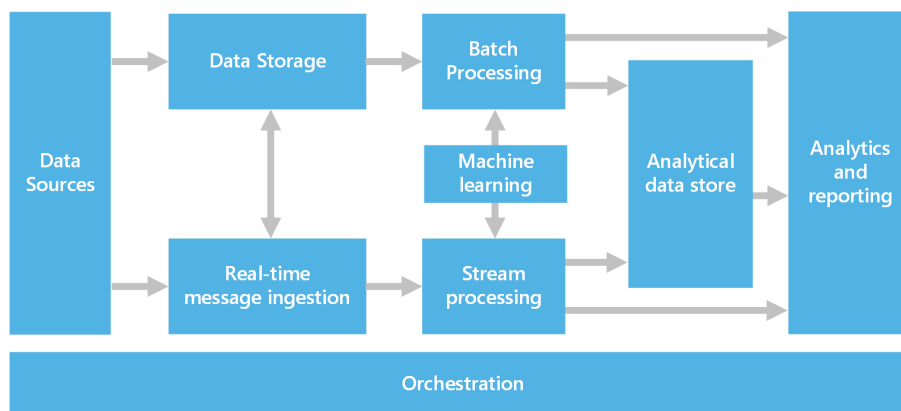


Fig. 1. A generalized model of BigData architecture according to [8]

visualization etc.).

According to the author most BigData solutions can be fitted into the above frame.

Over time several specific BigData architectures can be distinguished [8–10].

- (1) *The Lambda architecture (LA)* was first proposed by MARZ and WARREN [11]. It is based on three layers: “batch layer” (distributed file system) to store all data; “speed layer” to process data between two batch views in real time, and “serving layer” to index batch views for better querying. The LA is designed to minimize latency updates and maximize accuracy applications.
- (2) *The Kappa architecture (KA)* was developed by KREPS [12] in 2014 as alternative to Lambda architecture in order to simplify its structure. KA is based on two layers: “serving layer” for all inflow data and “speed layer” complete with reprocessing function. One of organizations using KA is LinkedIn founded by Kreps. A comparison between LA and KA is done in [13]. The results show better accuracy for LA with 9% at the expense of 2.2 times processing time and 10–20% of CPU usage.
- (3) *Internet of Things architecture (IoTA)*. A number of IoT architectures are known today [8, 14] (see Fig. 2).

IoT architectures are event-driven and they could have a huge number of applications and realizations everywhere where smart devices are used such as smart home, smart city, monitoring, security, etc.

- (4) *The Microservice architecture (MSA)*. Such architecture is efficient when a large number of services relatively weakly interconnected need to be performed. Each (micro)service is processed separately and independently [15]. Microservice architecture originated from Service oriented architecture (SOA) and they are often discussed without reason as opposed to each other. Microservice architecture is used in the activities of large organizations such as Amazon, eBay, Netflix.

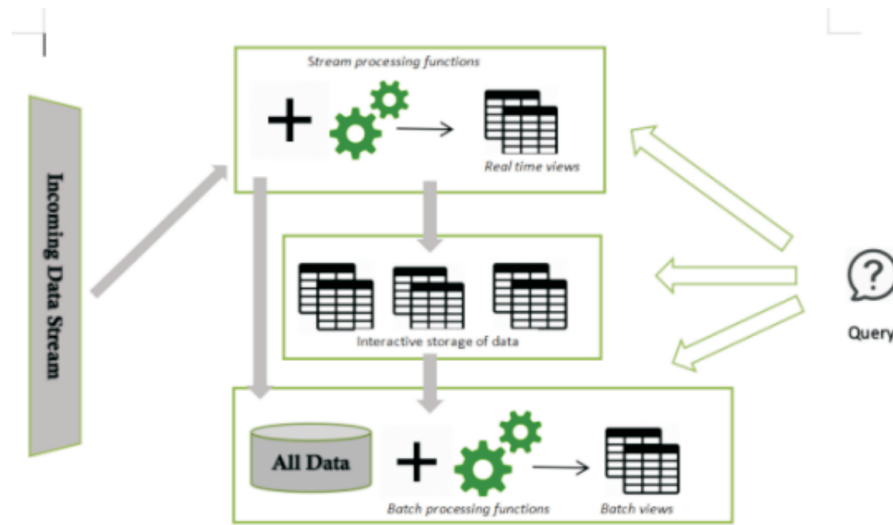


Fig. 2. An example of generalized Internet-of-Things architecture according to [13]

- (5) *The Zeta architecture (ZA)*. ZA is proposed by SCOTT [16]. It is an architecture closely related to the activities of an organization. Each activity is put into separate container where applications can be run separately. In this way a large number of (different) activities can be supported. As an example of Zeta architecture application is Gmail service of Google.

3. A model of BigData architecture. First, we formulate the main challenges that such an architecture must satisfy:

- The main question in the design is how to manage different types of context information in smart cities. These could be for example transport, information services, entertainment, security, social services, education, etc.
- The contextual information could come from many sources: other existing systems, users at different levels, through mobile applications and sensor networks.
- The systems involved in managing city services or third-party applications (subject to access control rules) can consume and/or produce contextual information. The entire city management can rely on all available contextual information (real-time and historical) to monitor and perform BigData analysis.
- A flexible approach is needed with the ability to integrate with existing or future systems without impacting their architectures.
- Information about the attributes of an object can come from different systems that act as either context producers or context providers. The source of information for an object may vary over time.
- Connecting to the Internet of Things: capturing data from or acting on IoT

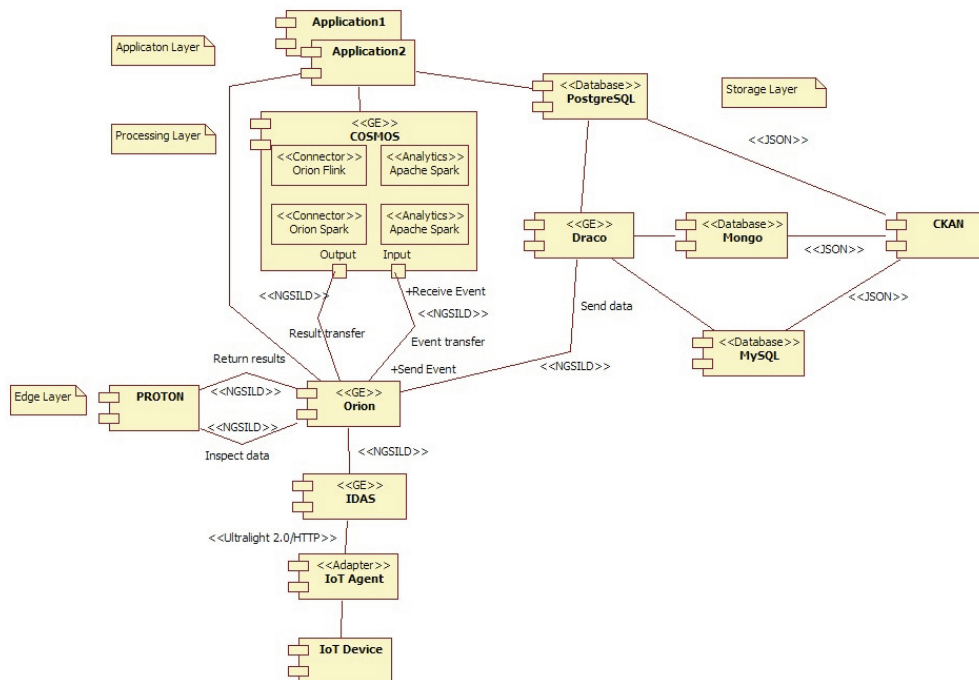


Fig. 3. A model of the new BigData architecture

devices as easily as possible using the Context Broker.

- The implementation and use of intelligent components and services to be easy.
- The architecture to be open source type.

By taking into account the above requirements the following model of BigData architecture is designed (see Fig. 3).

It consists of several main sections from bottom to top:

- Edge section
- Processing and storage section
- Application section

Each section is discussed in details below.

4. Discussion. The above proposed architecture is intended to be realized on the base of FIWARE components (see Fig. 3). According to [17] “FIWARE is an open alternative to existing proprietary Internet platforms”. Another question is how the proposed architecture itself, apart from the specific components from which it is constructed, relates to the above types of architectures. It can be said that the new architecture is a hybrid between the Lambda and Kappa architectures. It has two layers (edge and processing-and-storage) and the processing-and-storage layer has two sublayers that may operate independently or may be

interconnected. In the first case we have Lambda architecture, in the second case we have Kappa architecture.

The proposed model is a fully multi-functional BigData architecture. Our study shows that other developments based on FIWARE components are purposefully developed in close connection with the specific purpose [18,19].

In more details the components of the architecture are organized as follows.

Edge section itself consists of the following five components:

1. IoT device – any physical or software unit which can submit data or execute commands.
2. HTTP adapter. The role of adapter is to transfer data and commands from/to sensors to Ultralight 2.0 HTTP protocol. That protocol is one of the standard protocols supported by IoT agents.
3. IDAS is an IoT agent which transfers bidirectionally data/command coded in Ultralight 2.0 HTTP NGSILD protocol, which is standard interface between generic enablers (GE) included in FIWARE framework. Other adapters may support different protocols like ETSI M2M, MQTT, etc.
4. Orion is context broker. It is responsible for managing all the application context information modelled as NGSILD entities. Context information may be time, place, author, etc. of data gathered by IoT devices. It is the only Generic Enabler which is mandatory for any application claiming compatibility with FIWARE.
5. PROTON is a Generic Enabler whose role is to process context information according to previously created rules. The rules are defined by means of UX/UI supplemented with PROTON installation. For example, a rule may define an interval from allowable values of measurements of temperature sensor. When a measurement is out of the interval a NGSILD message to Orion context broker or COSMOS GE will be generated.

Processing-and-storage section consists of two separate sub-layers. They may operate independently or may be interconnected:

1. *Process sub section COSMOS GE*. Orion GE itself does not support durable storage. Context data acquired by Orion may be sent either to COSMOS GE for analytical processing or directly stored in database.

COSMOS GE consists of four main components: Apache Flink, Apache Spark, Orion Flink and Orion Spark.

- 1.1. Apache Spark is a unified analytics engine for large-scale data processing. It provides high-level APIs in Java, Scala, Python and R, and an optimized engine that supports general execution graphs. It also supports a rich set of high-level tools including Spark SQL for SQL and structured data processing, pandas (Python Data Analysis Library) API on Spark for pandas workloads, Mllib for machine learning, GraphX for graph processing, and Structured Streaming for incremental computation and stream processing.

1.2. Apache Flink ML is a library which provides machine learning (ML) APIs and infrastructures that simplify the building of ML pipelines. Users can implement ML algorithms with the standard ML APIs and further use these infrastructures to build ML pipelines for both training and inference jobs.

Apache Flink and Apache Spark are both capable to carry out stream and batch processing.

1.3 Orion Spark – connector.

1.4 Orion Flink – connector.

The processing engines Apache Spark and Apache Flink are connected to Orion via two connectors – Orion Spark and Orion Flink.

2. *Storage sub section* is composed of Draco GE and relational and/or NOSQL databases (Mongo). Draco is a GE allowing data coming from Orion to be stored on a durable storage. The necessity for using Draco is based on the fact that Orion does not contain historical data.

Application section. Applications are presented at the highest level. They communicate with databases, COSMOS or Draco. An application may use Wirecloud GE, which consists of a number of widgets called mashups. Mashups facilitate the development of operational dashboards which are highly customizable by end users.

5. Conclusion and future works. The intensive development of hardware and software tools provides many opportunities for solving various real-world tasks, such as working with large data sets. This, in turn, is a powerful incentive and opportunity for the development of different approaches for solving them [20].

In this paper a model of a multi-functional BigData architecture is proposed. This architecture is a hybrid between the Lambda and Kappa architectures. It has two layers (edge and processing-and-storage) and the processing-and-storage layer has two sublayers that may operate independently or may be interconnected. The model is built on the base of FIWARE components. The most important layer is the processing-and-storage one. The process tools are analytic tool Apache Spark and a library Apache Flink that are connected. The storage tool is organized by relational and NOSQL databases which are supplemented with Draco GE to ensure the ability to work with historical data. An important component of edge layer is the context broker Orion with the help of which the application context information is managed.

The future research will be focused in two main directions. The first one is testing the performance of the proposed architecture after prototype implementation. The second one is to apply the architecture for solving SmartCity problems.

REFERENCES

- [¹] OROZOVA D., I. POPCHEV (2020) Cyber-Physical-Social Systems for Big Data, Proceedings of the 21st International Symposium on Electrical Apparatus & Technologies (SIELA), Burgas, Bulgaria, 1–4, doi: 10.1109/SIELA49118.2020.9167161, <https://ieeexplore.ieee.org/document/9167161>.
- [²] OUSSOUS A., F. BENJELLOUN, A. A. LAHCEN, S. BELFKIH (2018) Big Data technologies: A survey, Journal of King Saud University, Computer and Information Sciences, **30**, 431–448. <http://dx.doi.org/10.1016/j.jksuci.2017.06.001>.
- [³] POPCHEV I., D. OROZOVA (2019) Towards Big Data Analytics in the E-learning Space, Cybernetics and Information Technologies, **19**(3), 16–25.
- [⁴] CHICALANOV A., M. LYUBENOVA, R. NIKOLOV (2016) Future of Internet, Ecosystem Services and Sustainable Regional Development, Proceedings of the 4th Annual International Conference on Ecology, Ecosystems and Climate Change, 11–14 July, 2016, Athens, Greece, ATINER'S Conference Paper Series, No: ECL2016-2135. URL Conference Papers Series: www.atiner.gr/papers.htm. ISSN: 2241-2891.
- [⁵] POPCHEV I., I. RADEVA (2019) Risk Analysis – an Instrument for Technology Selection, Engineering Sciences **LVI**(4), 5–20, ISSN:1312-5702 (Print), 2603-3542 (Online), DOI: 10.7546/EngSci.LVI.19.04.01.
- [⁶] KHAN M., M. UDDIN, N. GUPTA (2014) Seven V's of Big Data understanding Big Data to extract value, Proceedings of the 2014 Zone 1 Conference of the American Society for Engineering Education, 1–5, doi: 10.1109/ASEEZone1.2014.6820689.
- [⁷] OFFIA C., M. CROWE (2019) A theoretical exploration of data management and integration in organisation sectors, International Journal of Database Management Systems (IJDMs), **11**(1), 37–56, DOI: 10.5121/ijdms.2019.11103.
- [⁸] TEJADA Z. (2022) Big data architectures, Microsoft, <https://docs.microsoft.com/en-us/azure/architecture/data-guide/big-data/>
- [⁹] KALIPE K., R. K. BEHERA (2019) Big Data Architectures: A Detailed and Application Oriented Analysis, International Journal of Innovative Technology and Exploring Engineering (IJITEE), **8**(9), 2182–2190, ISSN: 2278-3075, DOI: 10.35940/ijitee.H7179.078919.
- [¹⁰] SINGH K., R. K. BEHERA, J. K. MANTRI (2019) Big Data Ecosystem: Review on Architectural Evolution. In: Emerging Technologies in Data Mining and Information Security, Advances in Intelligent Systems and Computing (eds Abraham A., Dutta P., Mandal J., Bhattacharya A., Dutta S.), **813**, Springer, Singapore, https://doi.org/10.1007/978-981-13-1498-8_30.
- [¹¹] MARZ N., J. WARREN (2015) Big Data: Principles and best practices of scalable real-time data systems, Manning, 330 pp, ISBN: 9781617290343.
- [¹²] KREPS J. (2014) Questioning the Lambda Architecture, O'Reilly.com, 2 July 2014, www.oreilly.com/ideas/questioning-the-lambda-architecture.
- [¹³] SANLA A., T. NUMNONTA (2019) A Comparative Performance of Real-time Big Data Analytic Architectures, IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC), July 2019, 1–5, doi: 10.1109/ICEIEC.2019.8784580.
- [¹⁴] HAUSENBLAS M. (2015) A modern IoT data processing toolbox, 24 Apr. 2015, https://fr.slideshare.net/Hadoop_Summit/a-modern-iot-data-processing-toolbox.

- [¹⁵] ALSHUQAYRAN N., N. ALI, R. EVANS (2016) A Systematic Mapping Study in Microservice Architecture, 2016 IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA), 44–51, doi: 10.1109/SOCA.2016.15.
- [¹⁶] SCOTT J. (2015) Zeta Architecture — What’s in the Name? <https://medium.com/the-ramp/zeta-architecture-what-s-in-the-name-b2f68b42b94a>.
- [¹⁷] FIWARE foundation, <https://digital-strategy.ec.europa.eu/en/news/fiware-foundation-research-those-digital-services-you-will-love>.
- [¹⁸] ALONSO A., A. POZO, J. CANTERA, F. DE LA VEGA, J. HIERRO (2018) Industrial Data Space Architecture Implementation Using FIWARE, Sensors (MDPI Basel), **18**(7), <https://doi.org/10.3390/s18072226>.
- [¹⁹] SALHOFER P. (2018) Evaluating the FIWARE Platform: A Case-Study on Implementing Smart Application with FIWARE, Proceedings of the 51st Hawaii International Conference on System Sciences, <http://hdl.handle.net/10125/50615>, doi: 10.24251/HICSS.2018.726.
- [²⁰] POPCHEV I., I. RADEVA, I. NIKOLOVA (2021) Aspects of the evolution from risk management to enterprise global risk management, Engineering sciences, **LVII**(1), 16–30, ISSN:1312-5702 (Print) 2603-3542 (Online), DOI:10.7546/EngSci LVIII.21.01.02, <http://es.ims.bg/indexx.htm>.

**University of Library Studies
and Information Technologies
119 Tsarigradsko Shosse Blvd
1784 Sofia, Bulgaria
e-mail: a.chikalanov@unibit.bg
e.kovatcheva@unibit.bg
r.nikolov@unibit.bg
e.shoikova@unibit.bg
l.gotsev@unibit.bg*

***Institute of Information
and Communication Technologies
Bulgarian Academy of Sciences
Akad. G. Bonchev St, Bl. 2
1113 Sofia, Bulgaria
e-mail: l_kirilov_8@abv.bg*

****Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Akad. Georgi Bonchev St, Bl. 8
1113 Sofia, Bulgaria
e-mail: al.iliev@math.bas.bg*