

WAZIUP: An Open Innovation Platform to Accelerate Innovation in Africa through cutting edge IoT cost effective communications and Big Data applications

Charlotte Dupont¹, Sabine Fatnassi¹, and Corentin Dupont²

¹ Create-Net via Alla Cascata, 56/D 38123 Trento, Italy

² Easy Global Market Espace Beethoven 1200 Route des lucioles 06560 Valbonne, France

`charlotte.dupont,sabrine.fatnassi@eglobalmark.com cdupont@create-net.org`

Abstract. Abstract...

1 Introduction

WAZIUP is a 3 years H2020 international cooperation action started on February 1st, 2016. The project is driven by a consortium of 5 EU partners and of 7 partners from 4 sub-Saharan African countries. Furthermore, it has support from multiple African stakeholders with the aim of defining new innovation space to advance the African Rural Economy. It will do so by involving end-users communities in the loop, namely rural African communities of selected pilots, and by involving relevant public bodies in the project development. WAZIUP will accelerate innovation in Africa by coupling with current EU innovation in the sector of IoT and Big Data: this EU technology will be specialized to generate African cost effective technologies with an eye to preparing the playground to the future technological waves by solving concrete current needs. WAZIUP will deliver a communication and big data application platform and generate locally the know how by training by use case and examples. The use of standards will help to create an interoperable platform, fully open source, oriented to radically new paradigms for innovative application/services delivery.

1.1 ICT African context

ICT, in Sub-Saharan Africa, in several cases has enabled convergence of productive sectors, serving as platform for more holistic development. In fact, there are many examples of ICT developments in Africa that cut across traditional sectors: notable examples are the introduction of micro-health insurance and health-savings accounts through mobile devices; index-based crop insurance; crowd-sourcing to monitor and manage the delivery of public services, etc. These innovative applications for several reasons more disruptive in social terms than

many counterparts in the EU - recognize and leverage commonalities between sectors, blur traditional lines, and open up a new field of opportunities.

The opportunity for ICT intervention in Africa is huge especially of IoT and big data: those technologies are promising a big wave of innovation for our daily life. It is widely accepted that the Era of IoT can potentially connect billions of sensors, devices, equipment, systems, etc. In turn, the challenge is about driving business outcomes, consumer benefits, and the creation of new value. The new mantras for the IoT Era are becoming collection, convergence and exploitation of data. The information collection involves data from sensors, devices, gateways, edge equipment and networks on to their respective siloed IoT platforms in order to increase process efficiency through automation while reducing downtime and improving people productivity.

WAZIUP targets the rural community in Sub-Saharan Africa because about 64% of the population is living outside cities. The region will be predominantly rural for at least another generation. The pace of urbanization here is slower compared to other continents, and the rural population is expected to grow until 2045. The majority of rural residents manage on less than few Euros per day. Rural development is particularly imperative in sub-Saharan Africa, where half of the rural people are depend on the agriculture/micro and small farm business, other half faces rare formal employment and pervasive unemployment. For rural development, technologies have to support several key application sectors like living quality, health, agriculture, climate changes, etc. WAZIUP project consider how to best design and deploy the IoT-Big Data technology considering cost and energy challenges in the first place.

Beside the cost and power consumption, the robustness of hardware is a core requirement: hardware has to be robust enough so as to require lower maintenance and handle environmental and deployment threats as well. WAZIUP will collect the grand challenge: reduce costs, reduce power consumption but at the same time increase the robustness of the hardware.

1.2 Project challenges

WAZIUP presents several challenges as listed below:

- *Innovative design of the IoT platform for the Rural Ecosystem.* Low-cost, generic building blocks for maximum adaptation to end-applications in the context of the rural economy in developing countries.
- *Network Management.* Facilitate IoT communication and network deployment. Lower cost solutions compared to state of the art technology: privilege price and single hop dedicated communication networks, energy autonomous, with low maintenance costs and long lasting operations.
- *Long distance.* Dynamic management of long range connectivity (e.g., cope with network & service fluctuations), provide devices identification, abstraction/virtualization of devices, communication and network resources optimization.
- *Big-data.* Exploit the potential of big-data applications in the specific rural context.

From a technical standpoint, WAZIUP introduces innovation by constructing on the following pillars of IoT/Big Data technology, specifically tailored for the rural ecosystem:

- *Privacy and security*: through attention to all related privacy and security aspects with specifics addressing the involved communities (farmers, developers);
- *Personalized and user friendliness*: models will receive requirements from users needs and will ensure compliance with all most common usability standards (e.g., Web Accessibility Initiative - WAI or ISO/TR 16982:2002);
- *An Open interoperable platform*: through open standard and protocols from the Geospatial Consortium (OGC), W3C, IEEE from the European SDOs (CEN, CENELEC and ETSI, etc.) for all its key technology
- *Continued Openness*: through the release of open specification and open software components and/or algorithms;
- *Low-cost and low-energy consumption*: through the design of innovation hardware (sensors/actuators), and of IoT communication & network infrastructure.

2 Related works

In [?], the author presents...

3 IoT & Big Data Platform

Challenges of WAZIUP will be tackled using an open IoT-big data platform with affordable sensors connected through an IoT-Cloud open platform. This platform will also make use of mobile phones and real-time processing to empower users and deliver the needed services. The project will not develop any new IoT/Big Data software platform, but, rather exploit the existing solutions and adapt them for the purpose. Hereafter a compact list of core technical functionalities encompassed by the platform:

- *Cloud-based real-time data collection combined with analytics and automation software*: thus, the platform will offer cost-effective solutions for aggregating different machines and sensor types to engender efficiency, smart automation and optimization in the rural context.
- *Intelligent analytics of sensor and device data*: studied in order to optimize for performance of the rural workplace, detect potential outages, and finally reduce overall maintenance costs.
- *Integration to 3rd parties' platform*: enables customers' benefit of scaling fast and easy.
- *PaaS (Platform-as-a-Service) provider*: WAZIUP will provide to business clientele with independently maintained platform upon which their web application, services and mobile applications can be built.

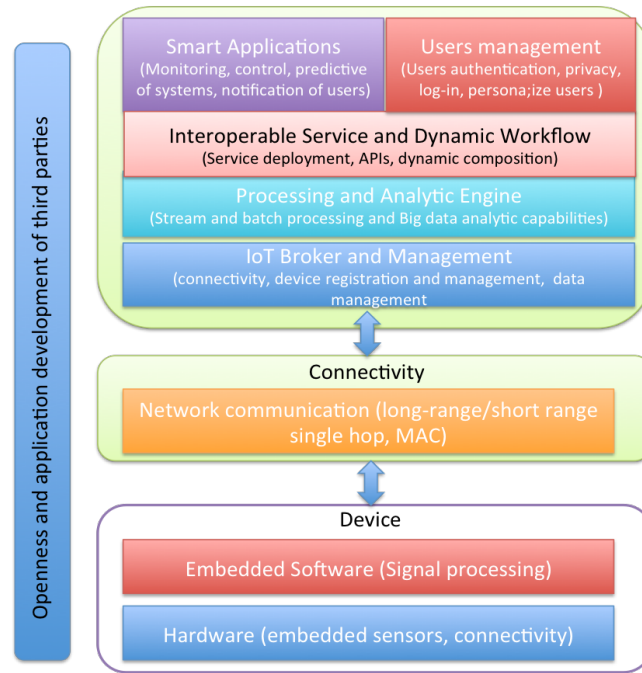


Fig. 1. Functional overview of WAZIUP

3.1 Platform overview

The Figure 3 displays the functional overview of WAZIUP. The topmost block represents the Cloud platform, the middle one is the network connectivity while the bottom one is the local deployment, including gateway and sensors.

3.2 IoT platform

IoT architecture overview Our iot platform architecture is presented in Figure 2. It has three main layers: device layer, gateway layer and the cloud layer.

- *The device layer* It includes IoT sensors and actuator that are able to communicate each on its protocol (MQTT, CoAP, HTTP, LoRAWAN) with the gateway. It a constraint and autonomous device. Considering the project circumstances it has to be low cost and doesnt consume much power.
- *The gateway layer* The gateway is a key component in our architecture since it ensure a secure bidirectional communication between the Iot devices and the cloud services. It interfere with various kind of devices: getting the data from sensors and sending commands to actuators. It has to be multi-protocol in order to be able to support diverse devices and it need to have device management capabilities. Since the connection is intermittent in our

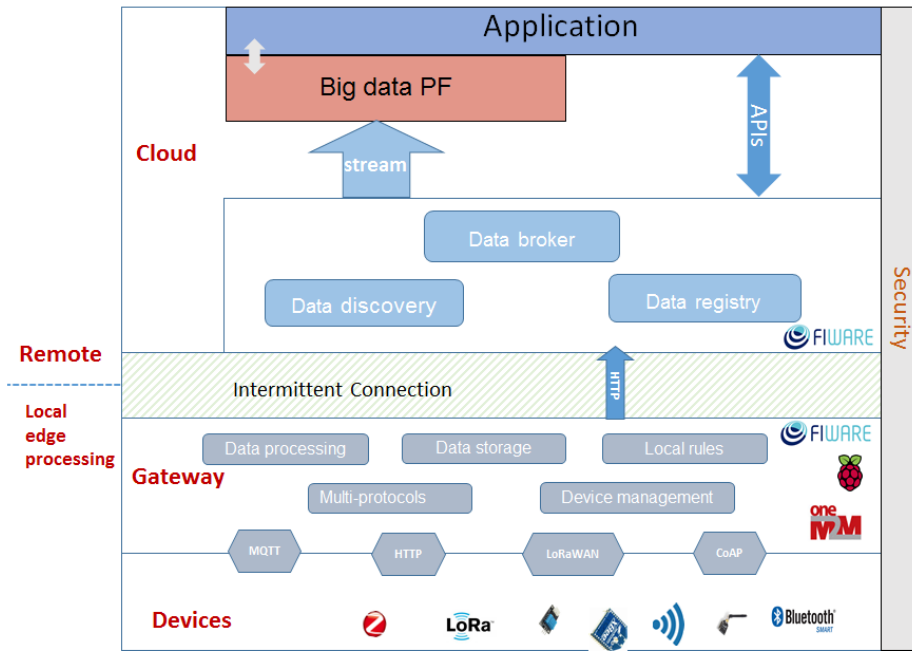


Fig. 2. IoT platform architecture

project, it is mandatory that the gateway is able to manage local rules (for example some event processing), to store data locally and to have low speed internet connectivity (2G: GPRS, EDGE).

- *The cloud layer* Our cloud main component are: the Data broker, the data registry and the data discovery.
- *Data broker*: it is the key component of the IoT cloud services, it is based on publish/ subscribe mechanism to ensure data transfer from different producers to their respected consumers.
- *Data registry*: it contains the virtual representation of devices (the data and the meta-data)
- *Data discovery*: it is responsible on querying the data.

Technology and tools In this section we will present the technology choices for each functional component stated above. In the cloud level the functional component are the data broker, data registry and data discovery. To cover these functionalities, The FIWARE generic enabler Orion context broker is the all-in-one solution. It is based on the Next Generic Service Interfaces (NGSI) 9/10. NGSI specification defines a data model and interfaces to manage the whole cycle of the virtual entities. We have also another candidate: the Nec IoT broker that cover the functionality of the data broker and that uses the ConfMan com-

ponent for the data discovery and registry. In the gateway level, we have these functionalities:

- *Multiprotocol* The FIWARE IoT agent component: can be an option to ensure this characteristic. Each IoT agent can be responsible on the data transfer to and from the devices using the device specific protocol.
- *Device management* We are considering the use of LwM2M device management protocol.
- *Local rules* FIWARE Cepheus enabler can be an option since it has a complex event processing engine.
- *Data processing* this component will be developed depending on use cases requirements.
- *Data storage* Mongo DB can be an option, other light data bases can be also considered such as SQLiteDB.

In the devices and networking level, LoRa technology captured our attention. It specifies LoRaWAN for Low Power Wide Area Network (LPWAN), the kind of technology needed in our project context (rural environment). Using LoRaWAN we can reach above 20 km in Line-Of-Sight (LOS) condition between devices and the gateway.

3.3 Big Data Platform

TODO ▶Charlotte◀

Big Data Platform overview

Review of Big Data tools **Corentin** ▶to be moved to related works?◀ Far from an exhaustive list, this paragraph describes the most used Open Source Big Data tools and compares them in order to give a better understanding of the Big Data ecosystem. Moreover, this review gives an indication on the best tools fitted for WAZIUP platform.

Databases and data warehouses HDFS, developed by Apache, is a distributed, scalable and portable file-system written in Java for the Hadoop Framework³. It has been designed for large dataset analysis and by its structure has high fault tolerance. It is the basis upon which everything works in the Hadoop Ecosystem. Build on top of HDFS, Apache HBase is a distributed, non-relational column oriented datastore⁴. HBase is designed to efficiently address random access and fast record lookup. It has the capability to handle extremely large tables of data with low latency. Though, this data storage tool should be used when random and real-time read/write access to data is needed and when many thousands of operation per seconds need to be performed on large datasets (up to petabytes).

³ https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html

⁴ <https://hbase.apache.org/>

Apache Hive⁵ is a data warehouse infrastructure that can manage and query unstructured data as if it were structured. As a full component of Hadoop Ecosystem, it uses MapReduce for execution and HDFS for storage. It has its own language SQL-like (HiveQL) that brings expressiveness to the queries. This storage mode should be used for SQL-like queries and when higher language than MapReduce is needed. Used by big companies who cant afford to lose data (Apple, Netflix, Spotify)(ref), Apache Cassandra is a column oriented database of structured data. The data are highly available thru column indexes and are automatically replicated thru multiple nodes for fault tolerance (ref). Cassandra has a unique masterless ring design that is easy to setup and to maintain (ref). This tools should be use when losing data is not the critical point and not affordable.

First considered as an outsider, getting rid of traditional table-based relational database, mongoDB quickly became a must-have tool: a NoSQL, relational, document oriented database (ref). Document are shared in JSON format with dynamic schemas (called BSON) and makes the integration of data sometimes easier. This database is very useful when you need to consume your data in many applications, as many connectors have been developed.

Data publication and subscription Apache Flume is a distributed, reliable and available service for efficiently collecting, aggregating and moving large amounts of streaming event data (ref). Flume should be used if the data is designed for Hadoop as it can move them to HDFS. It has many built-in sources and sinks and can process data in-flight using interceptors, which is useful for data masking or filtering. It is composed of agents and data collectors (and interceptors if needed).

More general purpose, Apache Kafka is a high-throughput, distributed, publish-subscribe messaging system (ref). It can replicate events, has low latency and is capable of data partitioning. Kafka is also easily scalable and this tool is very useful when the data is to be consumed by multiple applications. It is composed of producer, consumers and topics. Actually, we might not have to choose between Kafka and Flume, as both can work quite well together. If the workflow design requires streaming data from Kafka to Hadoop, using a Flume agent with Kafka source to read the data makes sense. This association is quite common and is called Flafka (ref).

If a Data/Context Scenario is developed, we may need to use a context broker. Orion Context Broker (FIWARE platform) is a publish/subscribe platform that is able to register context elements and manage them through updates and queries (ref). It is possible to subscribe to context information when some conditions occurs (e.g. an interval of time passed or the context elements have changed). Orion is a C++ implementation of the NGSI9/10 REST API binding developed as a part of the FIWARE platform.

Data processing Obviously, choosing a data processing tool depends mostly on the outcome expected from the data. The most common tool for BigData anal-

⁵ <https://hive.apache.org/>

ysis, and what we probably think at first, is Hadoop MapReduce. It has proven its efficiency in many ways and is an incredible tool. But if we want to step a bit aside of Hadoop workflow or if we have specific needs, other tools exists and some are becoming more and more powerful.

But first, lets talk about this milestone Hadoop MapReduce. MapReduce programming model contributed to the amazing progress of BigData processing this past decade (ref). By breaking down the work and recombining it in series of parallelizable operations, it is simple but incredibly efficient and scalable to ten thousands of machines if needed. It can run on inexpensive hardware, lowering the cost of a computing cluster. The latest version of MapReduce is YARN, called also MapReduce 2.0.

If a higher level of programming on top of MapReduce is needed, Apache Pig is the one. Pig has its own language (PigLatin) similar to SQL and works on top of MapReduce (ref). Pig Engine parses, optimizes and automatically executes PigLatin scripts as a series of MapReduce jobs on a Hadoop cluster. Its easy to learn and opens Hadoop to data professionals who may not be software engineers.

First designed to work with HDFS on top of YARN (ref), Apache Spark is a different system for processing data and can work out of Hadoop ecosystem with other data managements systems. It does not work with MapReduce and it can be up to a hundred times faster than MapReduce with its capacity to work in-memory, allowing to keep large working datasets in-memory between jobs, reducing considerably the latency (ref). What makes it more and more attractive to many users worldwide, is its wide range of applications: batch and stream processing (micro-batch processing with 0.5s latency), machine learning (MLib), SQL (with Hive), graph Analytics (graphX). Language supported are Java, Python and Scala.

Demand for stream processing becoming more and more important in Big Data analysis, Apache Flink has been recently developed (ref) and is growing very fast. Flink is a streaming dataflow engine that provides data distribution, communication and fault tolerance. It has almost no latency as the data are streamed in real-time (row by row). It runs on YARN and works with its own extended version of MapReduce. Language supported are Java and Scala.

Machine learning Machine learning is the union between statistics and artificial intelligence. It blends AI heuristics with advanced statistical analysis. We let the machine learn about the data, make decisions, and then apply statistics (ref). Algorithms used for this tasks can be grouped in 3 domains of actions: Classification, association and clustering (ref). To choose an algorithm, different parameters must be considered: scalability, robustness, transparency and proportionality. Overlearning (or overfitting) of the model must be carefully checked.

Without any math or programming requirement, KNIME is an analytic platform that allow the user to proceed the data in a user-friendly graphical interface (ref). It is a good tool to train your model and evaluate different machine learning algorithms rapidly. If the workflow is already deployed on Hadoop, a machine learning library exists and is called Mahout (ref). Spark also has his

own machine learning library called MLib (ref). H2O is a software dedicated to machine-learning, which can be deployed on Hadoop or Spark (Flink in development) (ref). It has an easy to use Web interface, which makes possible to combine big data analytics easily with machine learning algorithm to train models.

Data visualisation and exploration To visualise the data in real time, Freeboard provides a simple, real-time dashboard, commonly used in IoT world (ref). There is a direct Orion Fiware connector (ref). To connect with streaming engines, a JSON connector can be used. Design is simple and customisation is not possible, but it is a very good dashboard to visualise easily raw data coming from sensors, before data analysis.

Tableau Public offers a good visualisation and exploration tool on batch data. Tableau is a software where you can upload your analysed data (previously extracted in .csv format). The visualisation tool is very powerful and allow a deep exploration the data. However it is not designed for really Big Data with large datasets and the open Source version of Tableau (Public) does not offer the data streaming capacities (e.g. Spark connectors). Nevertheless, Tableau Public is a highly customisable, user-friendly and intuitive exploration tool for data that have already been processed and analysed.

To visualise data in real-time, after analysis (filtering, aggregating, correlating), one of the best tool is probably Kibana (ref). It is the visualisation tool coming with Elasticsearch. Elasticsearch is a search server based on Apache Lucene that provides a distributed, multitenant-capable full-text search engine with an HTTP web interface and schema-free JSON documents (ref). It is really designed for real-time analytics, most commonly used with Flink or Spark Streaming.

3.4 Platform as a Service (PaaS)

TODO ► *Corentin*◄

3.5 Deployment

Local and global Clouds An important feature in WAZIUP platform deployment is the possibility to have a local cloud platform. It is an infrastructure able to deliver services to clients in a limited geographical area. The local Cloud replicates some of the features provided by the traditional Cloud. It is used for clients that doesn't have a good access to the traditional Cloud, or to provide additional processing power to local services. In order for such an infrastructure to be considered as a Local Cloud it must support a virtualization technology. In the case of WAZIUP, the local Cloud comprises the end user or service provider PC and IoT Gateway. The local Cloud characteristics are:

- *Existence of IoT devices attached*
- *Can have geographical characteristics*
- *Must support virtualization*

- *Must support local cloud components*
- *Has an identifiable administrator/owner*
- *Has certain regulations/privacy considerations for data access and treatment*

The Global Cloud, on the other end, is a backbone infrastructure which increases the business opportunities for service providers and allows services to access a virtually infinite amount of computing resources. In order for such an infrastructure to be considered as a Global Cloud it must support a virtualization technology and be able to host the global cloud components of the WAZIUP architecture. The figure below present our deployment vision of WAZIUP platform:

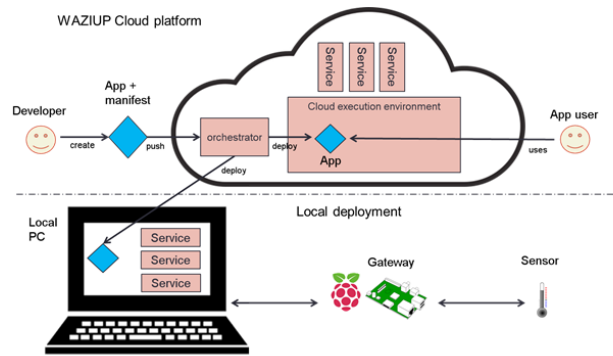


Fig. 3. WAZIUP deployment overview

The Figure 3 displays the WAZIUP deployment overview.

On the left hand side of the picture, the application is designed by the developer, together with the manifest file. It is pushed on the Waziup Cloud platform. The orchestrator then takes care of compiling and deploying the application in the various Cloud execution environments. Furthermore, the orchestrator drives the instantiation of the services in the Cloud, according to the manifest. The manifest is also describing which part of the application need to be installed locally, together with corresponding services. The local application can then connect to the gateway and collect data from the sensors.

Tools An interested tool for deploying code on the gateway level is resin.io. It facilitate the deploy, update, and the code maintain on remote devices. It is based docker containers and it supports several devices (the Rpi 1, 2 and 3 are supported).

4 African communities interactions

The project will build a strong community able to address innovation in the African rural ecosystem through EU ICT technology. Actors from various disciplines, including those having industrial backgrounds, those from application areas and target end-users will push towards the creation of an African-European innovation ecosystems based on the open sources for the IoT-Big data platform. Starting from the rural domain, the project technology will allow the African young application developer to produce new added value supporting multiple applications domains in the long run. The induced ecosystem renews the business paradigm: new benefits are engendered from providing and sharing devices, services, and data across stakeholders and applications.

4.1 Developers community

4.2 Rural community

4.3 Services providers community

5 Conclusion

References