QROWD - Because Big Data Integration is Humanly Possible

Innovation Action

Grant agreement no.: 732194

D8.1 – Requirements and Architecture

Due Date	30 May 2017
Actual Delivery Date	30 May 2017
Document Author/s	Semih Yumuşak (AI4BD) Daniel Hladky (AI4BD)
Version	0.7
Dissemination level	PU
Status	Final
Document approved by	

Project co-funded under Horizon 2020 Research and Innovation EU programme

TABLE OF CONTENT

Page

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

BDVA	Big Data Value Association
BDE	Big Data Europe
BD	Big Data
GE	Generic Enabler
WP	Work Package
API	Application Programming Interface
UI	User Interface

History

Version	Date	Reason	Revised by
0.0	01.03.2017	Initial version	Semih Yumuşak (AI4BD)
0.1	10.03.2017	Requirements elicitation and platform architecture	Semih Yumuşak (AI4BD)
0.2	27.03.2017	BDE and FIWARE explanations, Architectural enhancements	Semih Yumuşak (AI4BD)
0.3	03.04.2017	Corrections based on internal reviews	Semih Yumuşak Martin Voigt Daniel Hladky Erdoğan Doğdu (AI4BD)
0.4	26.04.2017	Introduction, summary, and process and approval	Semih Yumuşak (AI4BD)
0.5	26.04.2017	Review comments, Executive Summary, and corrections	Semih Yumuşak Daniel Hladky (AI4BD)
0.6	05.05.2017	Reviews and Corrections in Chapter 2	Mattia Zeni (UniTN) Luis-Daniel Ibáñez (SOTON) Pauline Baudens (TomTom)
0.7	16.05.2017	Revisions based on internal reviews	Semih Yumuşak (AI4BD)

EXECUTIVE SUMMARY

The D8.1 report addresses the functional and non-functional requirements leading to the QROWD platform architecture and design. The report incorporates the requirements from WP1 and WP2 but also from all other work packages. Within chapter 2 all architecture (A) and design (D) issues are summarized and are the base for the initial design and architecture incorporating the compliance of Big Data Europe (BDE) and FIWARE. All partners within D8.1 have actively contributed within the various on- and offline meetings. The initial high level architecture is well described in chapter 3 and 4 providing a clear insight into the functional process of the Data Value Chain. The proposed architecture and design explains how the various components of the work packages will be integrated, how they will interact and how they exchange data by using the proposed broker system that is compliant with BDE and FIWARE. The D8.1 report provides also an initial guide within chapter 5 explaining general guidelines for the development, acceptance and deployment assuring the QROWD compliance.

1 INTRODUCTION

In this document, the requirements analysis for the QROWD Platform is reported by means of both inputs from WP1 and WP2, resulting in a list of functional and non-functional requirements for the QROWD platform architecture and design. The requirements from other WPs were collected to design an interface to orchestrate the services listed under Section 2.

The high level architecture is defined in Section 3 to specify all functional and logical relationships among components to be developed in Data Value Chain WP4 - WP7 and QROWD crowdsourcing services in WP3. An abstract logical model is defined in Section 4 for the development of an ICT system able to properly manage all stages of the Data Value Chain in an integrated, coherent and consistent way.

This model will be supported by the definition of vocabularies and required communication protocol among components compliant with the FIWARE specifications. The architecture and design is based on the pre-requisite of being compliant with the BDV and FIWARE initiatives.

2 REQUIREMENTS ELICITATION

The requirements elicitation section spans into two categories: (1) Requirements from the Use Cases, (2) Requirements from the tool-centric WPs. The details of these categories are organized as architectural and design requirements for each WP. The summary of these requirements are listed in Section 2.3.

2.1 Requirements from the Use Cases

Work packages 1 and 2 are the main use cases to be studied under the QROWD project. The requirements for these WPs are listed in this section.

WP1- Business case: Advanced road information services for urban areas

In the first 12 months of this work package, a data collection and transformation task will be performed. The road information services datasets will be retrieved and converted from different sources. At the end of the project, the users of QROWD will be able to follow the analytical results.

Architectural Requirements

A1. InfAI will need an infrastructure for conversion and linking of the provided data by the use case. This requirement is directly related with WP5.

A2. UniTN will need data curation tools for this use-case, which is also related with WP7 requirements.

A3. In order to run the machine learning methods on datasets and floating data, InfAI will need a processing infrastructure. This requirement is also related with WP6.

Design Requirements
D1. SOTON will need an UI to follow and support the integration task.
D2. Partner UIs are needed in the QROWD platform to follow the data flow and the analytics.

WP2- Business Case: Intelligent urban transportation and mobility

In this case, the requirements for the work package will be collected in the first 12 months together with an ideas competition.

Architectural Requirements
A4. UniTN will need processing infrastructure for the collection and evaluation of urban mobility services.
A5. At this stage we see Apache Cassandra[1] and Apache Spark[2] as the two main components needed.

Design Requirements
D3. The ideas competition will need the QROWD platform to invite panel members and conduct competition.
D4. A dashboard for data analytics is needed.

2.2 Requirements from the Tool-Centric WPs

WPs from 3 to 8 focus on developing tools for the QROWD platform. In this section, the requirements of these work packages are listed.

WP3- QROWD crowdsourcing

In this work package, first, we aim to design a framework to effectively support stakeholder participation and collaboration along the entire cross-sectoral urban Big Data Value Chain in transportation and mobility. Second, we provide a unified interface to a broad set of crowdsourcing services including paid microtasks, niche sourcing (crowdsourcing with expert crowds) and gamification.

Architectural Requirements
A6. In order to deploy the crowdsourcing services and the related endpoints to support other work packages, SOTON will need an infrastructure for the crowdsourcing middleware (e.g. CrowdFlower[3], PYBOSSA[4]) and an interface enabling each of the other components to start and finish a crowdsourcing task. This entails the requirement of waiting for a crowdsourcing task to be finished before continue with the workflow. This tool also needs the ability to modify data stored in the platform.

[1] http://cassandra.apache.org/
[2] http://spark.apache.org/
[3] https://www.crowdflower.com/
[4] http://pybossa.com/

Design Requirements

D5. SOTON will need a management UI in the QROWD platform to perform crowdsourcing tasks, in terms of task and time management, and public endpoints. This UI will support the inflow of crowdsourced data into the QROWD system.

WP4- Hybrid data generation and acquisition

In order to perform data generation and acquisition tasks, following requirements are listed.

Architectural Requirements

A7. In order to store data in different formats, there will be a need for a data storage framework.
A8. In order to adapt data flows (such as RSS or crowdsourcing results) into QROWD, there will be a need for a data processing server and storage.
A9. The linked data generation task will require a processing infrastructure, possibly Karma Data Modeling tool to be deployed.
A10. For the crowdsourced multilingual data harvesting task, NLP frameworks (FOX[5], MINER) are required to run in the QROWD platform.

Design Requirements

D6. A final conclusion needs to be made if Karma modelling tool has to be adapted into the QROWD platform. In the current design stage we include it in the initial stage.
D7. Dashboards needed for FOX and Miner NLP frameworks, as well as and simplified UIs for annotation and verification tasks.

WP5- Hybrid link discovery and fusion

In this work package, the frameworks required to be deployed and the design requirements are listed below.

Architectural Requirements

A11. A deployment of crowdsourcing enabled LIMES framework is required for the link discovery
A12. DEER framework is required for data fusion
A13. The frameworks deployed in this package require access to the data from storage and need to store the links back to the storage.

WP6- Hybrid data analytics

The data analytics work package requires integration with the data collected by other work packages.

Architectural Requirements

[55] http://aksw.org/Projects/FOX.html

A14. For the real time data analytics, DL-Learner project will be enhanced to support crowdsourcing tasks; this tool will be required to be integrated under the QROWD platform.
A15. AI4BD requires a processing infrastructure for integrated processing of the data-in-motion and data-at-rest prototype.

Design Requirements
D8. Data Analytics dashboard and management UI is required in the QROWD platform.

WP7- Data storage and hybrid curation

For the data storage and curation tasks, the required architectural and design requirements are listed below.

Architectural Requirements
A16. For the link crowd-enabled quality assessment task, InfAI requires a processing infrastructure for RDFUnit and ORE tools.
A17. For the acquisition, storage and access of streaming data, UniTN requires Apache Cassandra[6] as a data storage solution and Apache Spark[7] as computational framework (both BDE compliant)
A18. For the data integration, storage and access, the QROWD platform requires a data storage solution (preferably from BDE)

Design Requirements
D9. For NER tasks, AI4BD will develop a crowd-enabled UI integrated in QROWD.

WP8- QROWD Platform

The QROWD platform is required to be based on the BDE[8] infrastructure and to be compliant with FIWARE[9]. Together with that infrastructure, all other work packages are required to be integrated or compliant with the QROWD platform.

The requirements to comply with these are listed below.

Architectural Requirements
A19. In order to run the BDE infrastructure, following Docker services are required to be up and running.
- Docker Machine
- Docker Engine
- Docker Swarm
- Docker-Compose

A20. FIWARE compliance of the platform is required, which means a

[6] http://cassandra.apache.org/
[7] http://spark.apache.org/
[8] https://www.big-data-europe.eu/
[9] https://www.fiware.org/

compliance with the FIWARE APIs and GEs[10]. Specifically, FIWARE Orion Context Broker is required to provide the compliance with the other working FIWARE compliant systems and projects.

A21. In order to provide data storage for different data formats, a format independent storage is needed. FIWARE CKAN platform provides a platform for this purpose. HDFS can also be used for this purpose.

Design Requirements

D10. The QROWD platform will include many components both in the infrastructure and at the UI level. These components are required to be scalable for BD platforms. The platform is also required to support integration of different modules, UIs, and APIs.

D11. The QROWD platform is required to provide a way to manage the infrastructure.

D12. It will allow the user to manage the data flow from data collection to data analytics. The user roles for the QROWD platform is listed below.

- **Data scientist:** how setup the data flows, e.g., data acquisition, storage, linking with crowdsourcing, … and runs the data flows → the main role
- **Developer / admin:** creates / bundles new service, deploy them for the data scientists
- **User / workers**: in the "crowd" are people who use the task UIs (web apps, mobile apps, …) to execute the crowd tasks
- **Use-case users:** who use the crowd-curated data, e.g., in use case applications or in dashboard visualizations

2.3 Summary of Requirements

The requirements are grouped in Table 1.

Table 1 : Summary of Requirements

Functionality	Required by	Requirements	Relevant components
Crowdsourcing Services	WP3	A6, D5	CrowdFlower, PYBOSSA
Crowd-sourcing enabled data interlinking and Fusion	WP1,WP5, WP4, WP7	A1, A10, A16, D1, D7, A11, A12, D9	Crowdsourcing-enabled LIMES and DEER, MINER, FOX
Linked Data Generation	WP4	A9, D6	Karma
Data Acquisition	WP4	A4, A5, A17	Apache Cassandra, Apache Spark
Data Curation	WP1,WP7	A2, A16	RDFUnit, ORE
Data Analytics	WP1, WP6	A3, A14, D2,	DL-Learner

[10] https://www.fiware.org/developers-entrepreneurs/

		D4, D8	
Crowdsourcing	WP3	A6, D5	Crowdsourcing services
Data Storage	WP4	A7, A8, A13, A18	Apache Cassandra, BDE Storage Solutions (HDFS, RDF Stores etc.)
Stream Processing	WP6	A15	Apache Kafka
Infrastructure	WP8	A19, A20, A21, D10, D11, D12	Docker Services, BDE, Fiware

2.4 The role of Big Data Europe and FIWARE for the Integration Platform

Big Data Europe (BDE) is a project aiming to make Big Data based systems simpler, cheaper and more flexible (Big Data Europe General Platform Description, 2016). BDE provides pipelines and sample applications stack based on Docker (Docker Web Site, 2017). Docker Swarm allows BDE apps to run in a scalable environment, whereas developed as stand-alone services. In Figure 1. BDE Platform Infrastructure , a high-level abstraction of the platform is illustrated. In the QROWD infrastructure, exact BDE infrastructural patterns that will be used is described as Docker Swarm and Docker Compose application bundles.

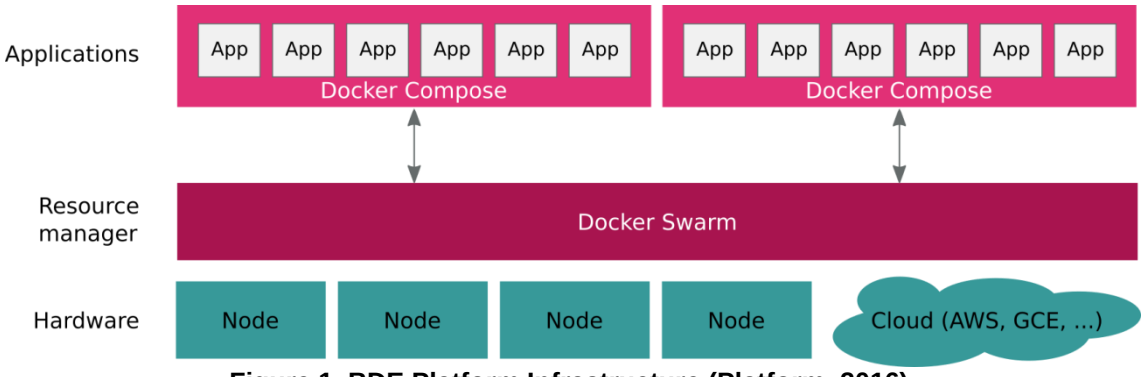

Figure 1. BDE Platform Infrastructure (Platform, 2016)

Applications are bundled on the BDE platform using pipelines, which can be defined by using a docker compose file. These files are named as "docker-compose.yml" and defines the configurations inside the pipeline. A sample BDE pipeline can be created as:

```
version: '2'
services:
  app:
    image: bde2020/pipeline-builder-frontend:0.4.2
    links:
      - identifier:backend
    ports:
      - 80:80
  identifier:
    image: semtech/mu-identifier:1.0.0
  dispatcher:
    image: semtech/mu-dispatcher:1.0.1
    volumes:
      - ./config:/config
  database:
    image: tenforce/virtuoso:1.0.0-virtuoso7.2.4
    environment:
      SPARQL_UPDATE: "true"
    ports:
      - "8890:8890"
    volumes:
      - ./data/db:/data
  pipeline:
    image: bde2020/mu-pipeline-service:0.1.0
  export:
    image: bde2020/mu-pipeline-export-service:0.1.0
```

Figure 2. Sample BDE Pipeline (Pauwels, 2016)

In the QROWD platform, core BDE platform (Docker Swarm, Docker Compose) will be used as the base, together with several apps listed in the BDE application stack (Kafka, Elasticsearch, DEER, Limes, Virtuoso, Hadoop, Spark, etc.). In order to position this new BDE-based QROWD platform as integrable with several domain specific applications and platforms, FIWARE apps will also be included in the pipeline.

In more detail, FIWARE (FI-WARE Consortium, 2012) is a middleware platform, and it is claimed to be the development and global deployment environment for applications in Future Internet[11]. FIWARE provides an enhanced OpenStack-based cloud environment[12] plus a rich set of open standard APIs that make it easier to connect to the Internet of Things, process and analyze Big data and real-time media or incorporate advanced features for user interaction.

Together with many tooling GE's developed or enhanced for the FIWARE platform, FIWARE has a core Broker GE, which is called Orion Context Broker. The idea of Context Broker GE is illustrated in Figure 3. Context Broker Generic Enabler

[11] https://en.wikipedia.org/wiki/FIWARE
[12] https://www.fiware.org/developers-entrepreneurs/

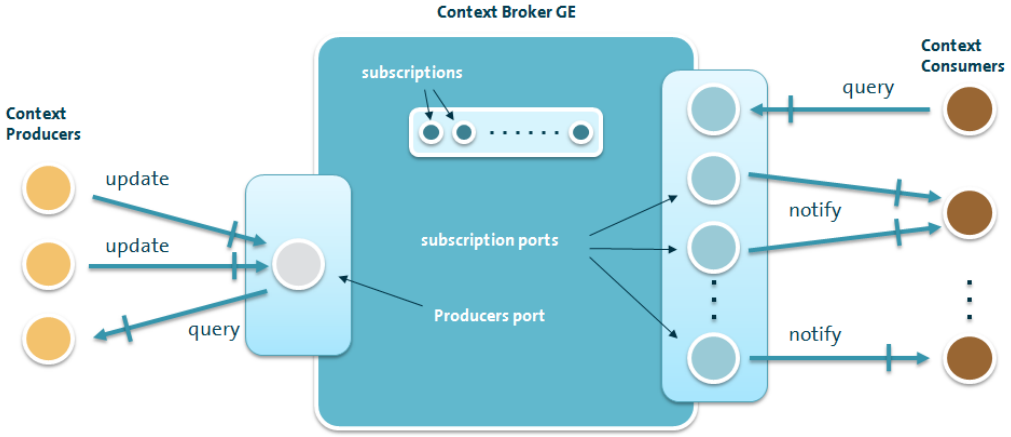

Figure 3. Context Broker Generic Enabler (Telefonica I+D, 2016)

With the inclusion of the Context Broker GE in the QROWD platform, the platform will be compliant with FIWARE specifications[13]. Thus, the QROWD platform will be integrated to any application supporting FIWARE specifications.

In terms of Big Data characteristics, BDE and FIWARE platforms are compared with some of the available Hadoop distributions in

Table 2 : Enhanced for Linked Data from. Whereas BDE platform provides a dockerized infrastructure to run Big Data applications in a scalable environment, FIWARE provides a Context-Based rich API to integrate many different platforms.

Table 2 : Enhanced for Linked Data from (BDE Architecture, 2016)

	Hortonworks	**Cloudera**	**MapR**	**Bigtop**	**BDE**	**FIWARE**
File System	HDFS	HDFS	HDFS	HDFS	HDFS	HDFS + Cloud
Installation	Native	Native	Native	Native	Light weight virtualization	Native
Plug&Play Components	no	no	no	no	yes	Yes
High Availability	Single Failure recovery (yarn)	Single Failure recovery (yarn)	Self healing, mult. Failure rec.	Single Failure recovery (yarn)	Multiple Failure recovery	Multiple Failure recovery

13 https://forge.fiware.org/plugins/mediawiki/wiki/fiware/index.php/Summary_of_FIWARE_Open_Specifications

Cost	Commercial	Commercial	Commercial		Free	Free
Scaling	Freemium	Freemium	Freemium		Free	Free
Addition of custom components	Not easy	No	No	No	Yes	Yes
Integration Testing	yes	yes	yes	yes	Yes	yes
Operating Systems	Linux	Linux	Linux	Linux	All	Linux
Management tool	Ambari	Cloudera Manager	MapR Control System	-	Docker swarm UI+ Custom	FIWARE Cloud
Semantic Web Support	No	No	No	No	Yes	No
API Support	No	No	No	No	Yes	Context-Based rich APIs
Open Data Publishing Tool	No	No	No	No	Yes	CKAN
SPARQL Endpoint	No	No	No	No	yes	No
Virtualization	No	No	No	No	Docker	FIWARE Cloud

3 GENERAL ARCHITECTURE

Based on the initial design and the requirements, QROWD platform will consist of a BDE-like applications stack, which will consist of several data pipelines. In this section, the details of this stack are explained in terms of data flows between different work packages and services.

In Figure 4: Data Flow, the data flow between work packages are summarized. All work packages need different levels of data integration, which are marked with different types of arrows defined below:

- Direct Integration: Integration between components by using component-specific interfaces.
- Data Integration via Broker: Integration between components by using a middleware broker such as Kafka or Orion Context Broker.
- Static Data Exchange: One-time static data transfer between different work packages.

As illustrated in Figure 4: Data Flow, there will be direct integrations, broker integrations, and static data exchanges between work packages from 3 to 7.

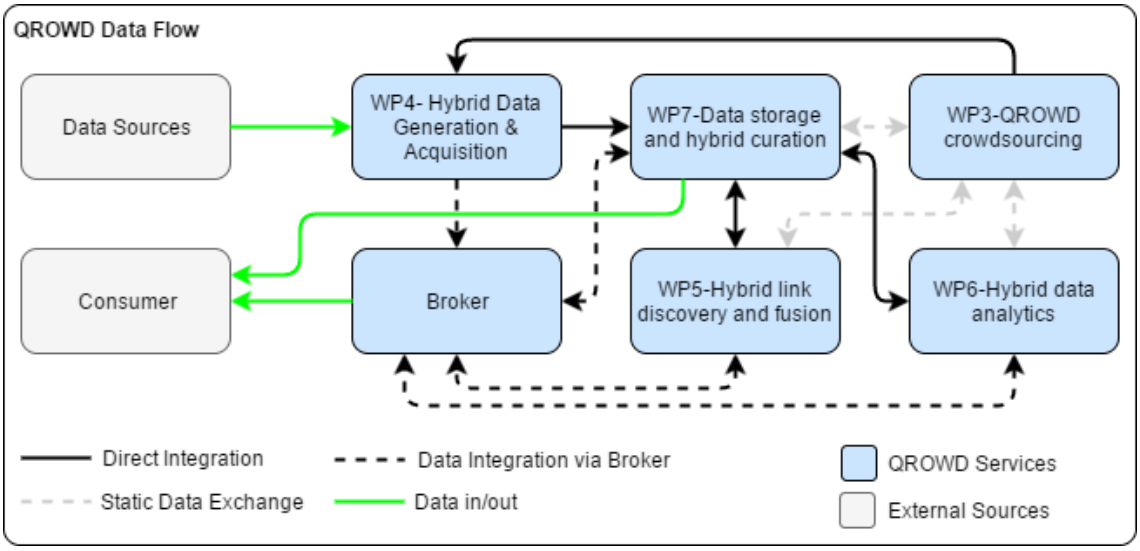

Figure 4: Data Flow Between QROWD components

From the BDE point of view, some of these data flows will be maintained by using the BDE platform (Jabeen, 2016). BDE components such as Hadoop, Elasticsearch, Kafka, Limes, DEER, etc. will be used as the base applications for the QROWD platform.

As a BDE applications stack illustrated before in Figure 1. BDE Platform Infrastructure , a pipeline will be created for QROWD applications. As a preliminary design, we have created a sample BDE-like applications stack , illustrated in Figure 5: Sample BDE.

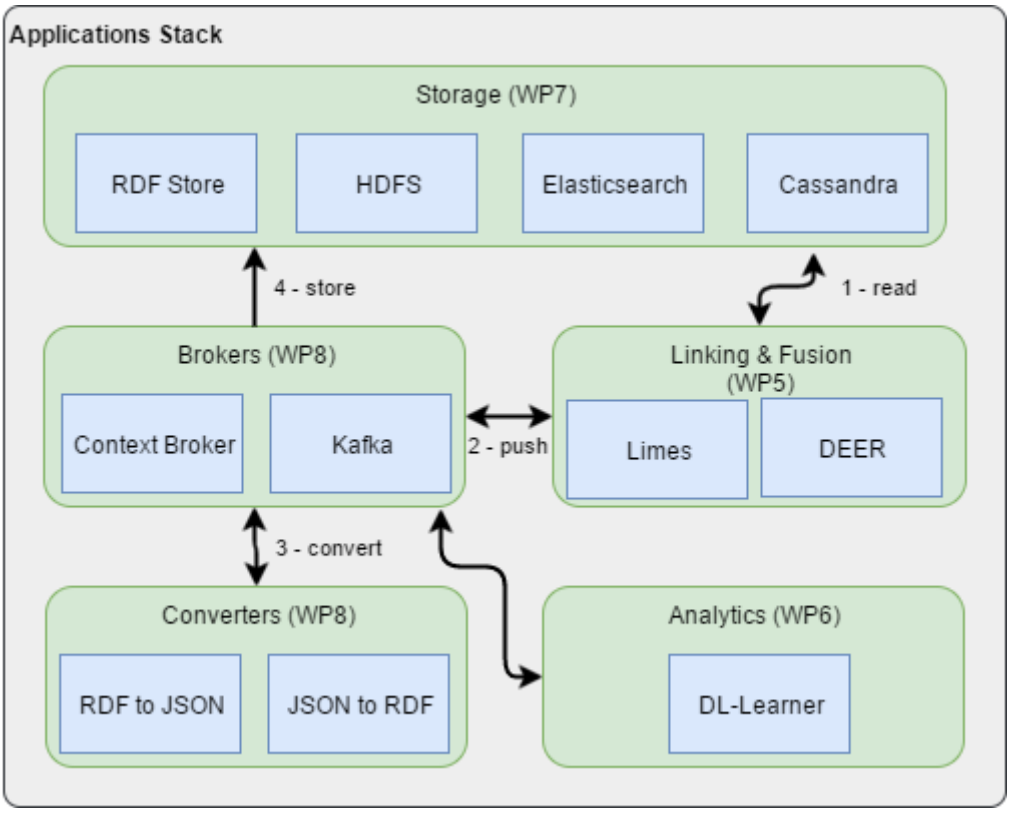

Figure 5: Sample BDE-like Applications Stack

Based on this design, data will be read, pushed, converted and stored between different services, whereas different applications are created in different work packages. As a higher level abstraction for the data flow, the complete architecture for the QROWD platform is illustrated in Figure 6: QROWD Architecture.

The data in-flow for the QROWD platform will be started by WP3 and WP4, then will be processed and stored by processes under WP5-7. QROWD platform, which will be developed in WP8, will include UIs, APIs and services that will provide the required infrastructure.
In Figure 6: QROWD Architecture, more detailed and a fuller picture of the QROWD platform is illustrated.

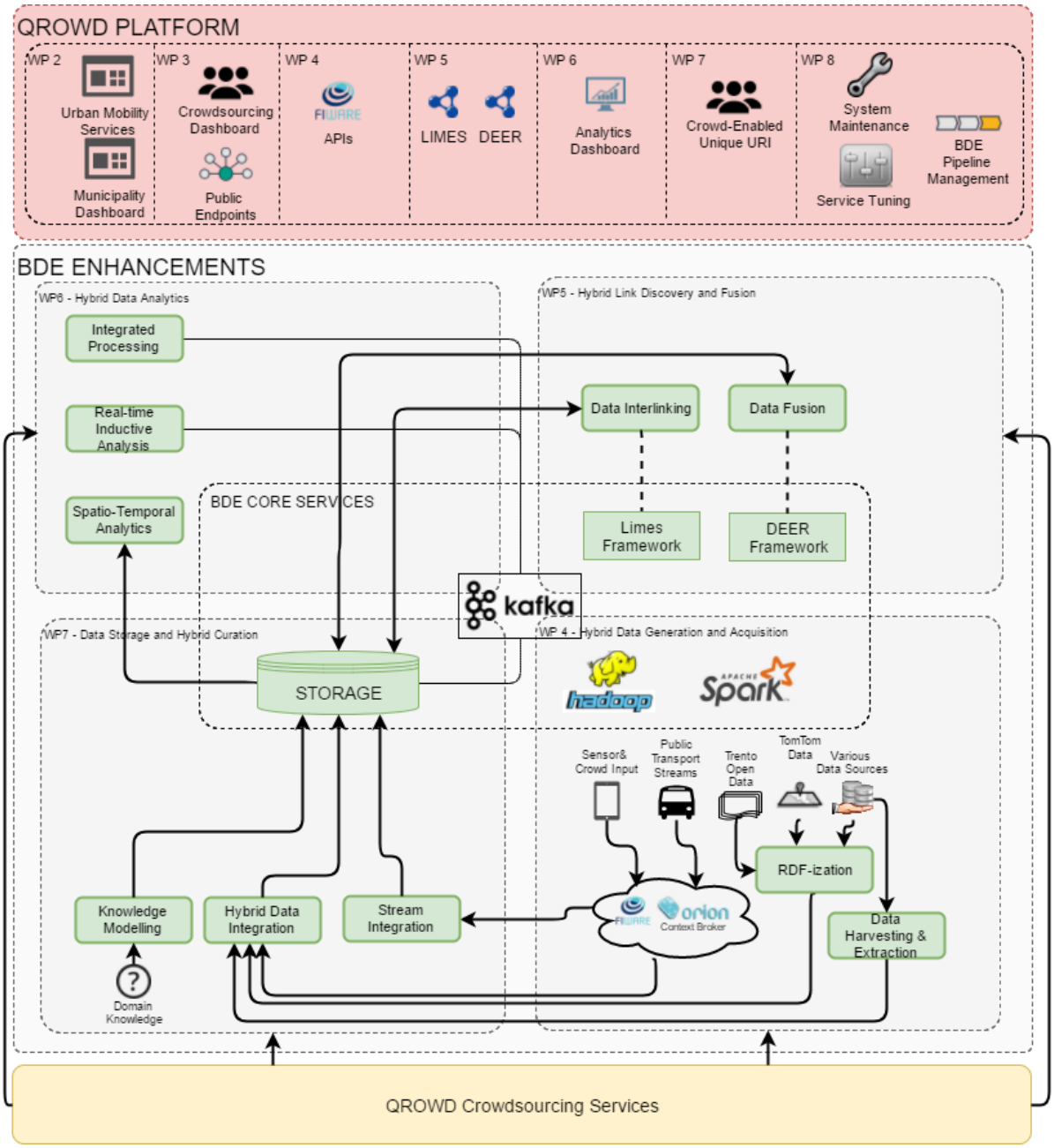

Figure 6: QROWD Architecture

As illustrated in Figure 6: QROWD Architecture, all applications developed in WP4-7 will be defined in the BDE pipeline. The core BDE services will include Kafka, Hadoop and Spark deployments, as well as a selected Storage server and NLP framework applications. As a inherent requirement for the BDE base, all applications will be Dockerized and run as Docker containers.

The FIWARE compliancy will be provided through FIWARE cloud applications (i.e. Orion Context Broker).

During the development of different Work Packages, QROWD platform will be used as an integration platform. Different BDE based dockerized applications

will be run together in an integrated BDE pipeline and will be represented by UIs supporting the data flow and reporting.

4 SPECIFICATION OF THE TOOL PACKAGING

BDVA is a non-profit organization aiming to boost research and development processes for Big Data. BDVA provides a reference architecture for Big Data systems to assist projects to have standardized infrastructures. The QROWD platform will be compliant with the BDVA reference architecture, which is shown in Figure 7.

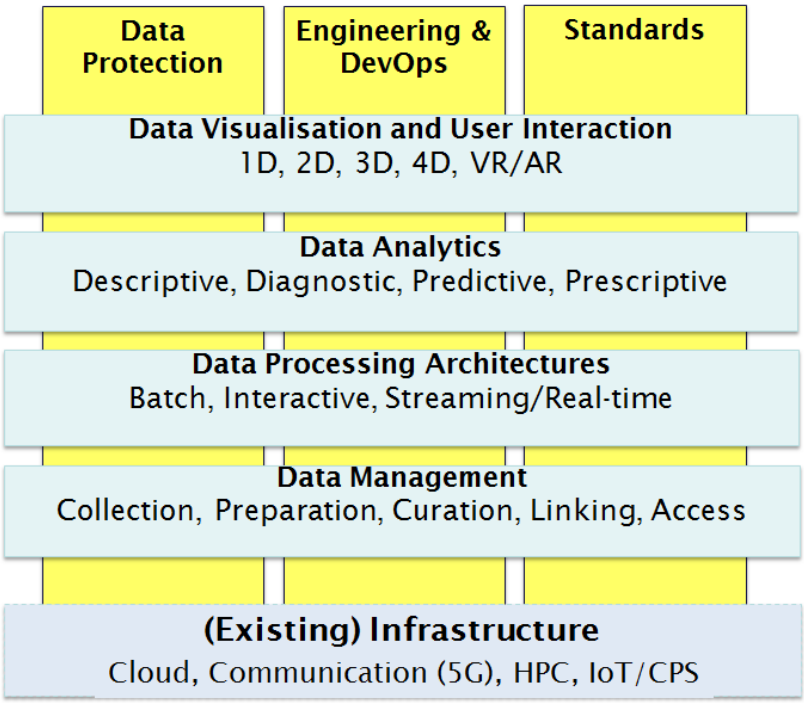

Figure 7: BDVA Reference Architecture

Based on the reference architecture, QROWD platform will have the following infrastructure illustrated in Figure 8: QROWD BDVA Infrastructure, which include applications developed in different work packages.

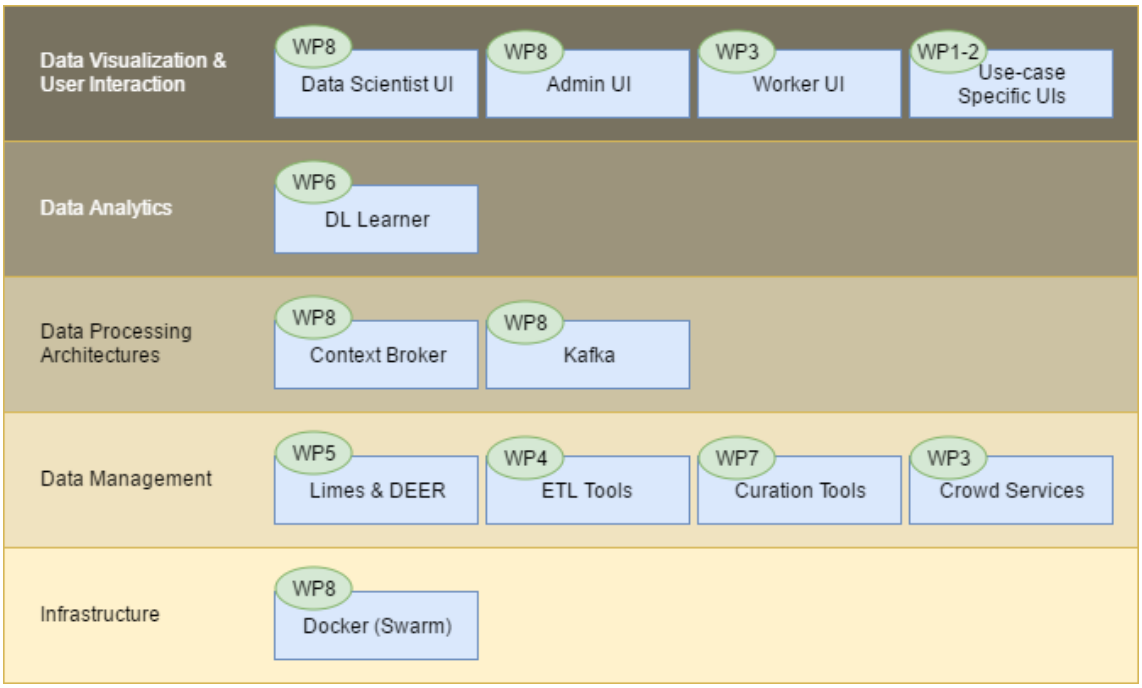

Figure 8: QROWD BDVA Infrastructure

5 DEVELOPMENT PROCESS AND APPROVAL

Development of QROWD components requires an integration testing and an approval process. Components to be integrated into the QROWD platform has to follow the rules defined in this section. The integration patterns to be followed is explained below.

5.1 Integration patterns:

The component owners will define integration patterns based on the following integration types.

5.1.1 API Integrations:

Inside the QROWD project, there will be 3 classes of APIs developed:

- Internal APIs: In every work package, there will be internal APIs to be used within the work package. REST, SOAP/HTTP, or web services will be developed as internal APIs. These APIs are not required to be shared between work packages.

- External APIs: External APIs will be available externally to consumers and partners. These APIs will be developed as REST/JSON for individual developers and web services (e.g. WSDL) for industrial integrations. These APIs are required to be compliant FIWARE NGSI specifications.

- Linked Data/SPARQL APIs: These APIs will integrate Linked Data based applications both within the QROWD project and with external Linked Data

applications.

5.1.2 Message Broker Integrations:

Big data integrations may require a message broker between different services to ensure the scalability of the data transfer. A message broker is an architectural pattern for message validation, transformation and routing (Gregor Hohpe, 2011). The message brokers will be positioned in two different aspects:

- **Enterprise application integrations:** In this context, Apache Kafka[14] is recommended as a pub-sub messaging intermediary. In the QROWD platform, BDE Apache Kafka solution will be configured and become available to other work packages as an integration tool. In order to integrate services through Apache Kafka, developers will need to request access to the broker and share their data schema.

- **FIWARE compliant integrations:** In order to provide FIWARE compliancy within the QROWD platform, FIWARE Orion Context Broker[15] will be served as an integration tool between other FIWARE compliant systems.

5.1.3 Direct Integrations:

All work packages may create direct integrations (such as database access, file sharing, or socket streams) to other QROWD modules. In order to create this direct integration within different modules, the developers will ask for permissions from the module administrator. Although direct integration will be supported by the platform, it is not recommended as the best option. The component developers are recommended to use API or Broker integrations as default.

5.2 Testing and approval:

A component developed to be run on the QROWD platform will be tested and approved based on the following rules of approval.

As explained in section 2.4, BDE infrastructure is based on Dockerized services. In order to comply with the BDE standards, the modules developed under WP3-7 is required to be dockerized. The process can be summarized as below;

(1) Check and run if there is a ready to run docker image.
(2) If not, create a customized Docker image.
(3) Document it.

These Dockerization steps are explained in detail in Figure 9: Running and Integration of a Docker Engine.

[14] https://kafka.apache.org/
[15] https://catalogue.fiware.org/enablers/publishsubscribe-context-broker-orion-context-broker

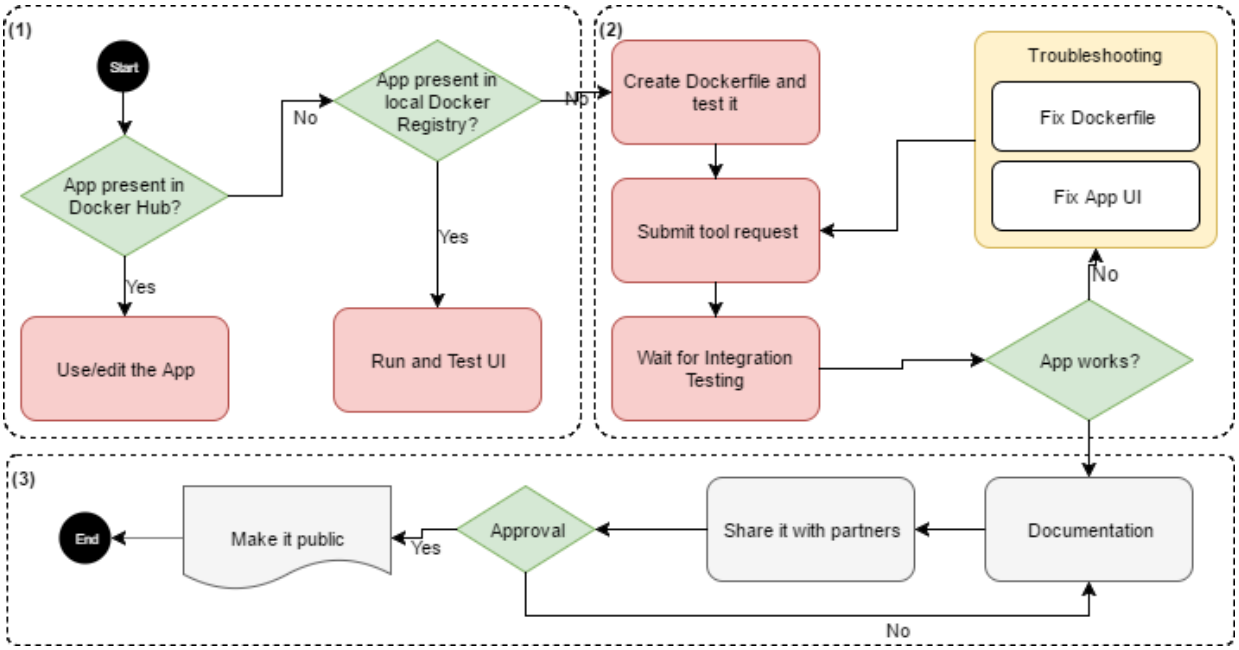

Figure 9: Running and Integration of a Docker Engine

6 CONCLUSIONS

In this document, the QROWD platform Big Data Value Chain is explained by means of the infrastructure and the module integrations. For each stage of the Big Data Value Chain, the project will deliver one or more components suitable to support the related tasks by means of crowdsourcing services. A framework will be defined and implemented to enable creation, composition and integration of services supporting the Big Data Value Chain, by providing enhanced OpenStack-based cloud hosting capabilities and components offering functions offered “as a Service”.

Based on the specifications and integration patterns defined in this document, the QROWD platform integration will mediate data integration between different work packages and different modules. The platform will be developed based on the BDE infrastructure, which will run BDE services and customized Docker images.

This report further provides a process guideline for all contributing parties in order to develop/deploy modules within the QROWD platform and coordinate/support QROWD services implementation in order to follow the defined specifications.

This QROWD platform will ensure FIWARE-compliance of QROWD services as required by the business cases. The BDE Platform will be used to support the services deployment in the cloud, which will simplify the integration by the use cases. In order to provide FIWARE compliancy, the platform will be integrated with FIWARE core services. The proposed architecture and design has incorporated all requirements and is further compliant with BDE and FIWARE specifications.

For the stand-alone QROWD Platform, an interface for orchestrating the QROWD services will be implemented. Two releases are envisaged, in order to allow the early usage of the platform for the pilots (WP1, WP2).

REFERENCES

BDE Architecture. Ertico Network. 2016-09. http://erticonetwork.com/wp-content/uploads/2016/09/Hajira-Jabeen-BDE-Architecture.pdf (accessed 2017-03-17).

Big Data Europe General Platform Description. https://www.big-data-europe.eu/wp-content/uploads/BDE-TECHFLYER-WEB-1.pdf (accessed 2017-03-17).

Docker Web Site. Docker Inc. 2017. https://www.docker.com/ (accessed 2017-03-17).

FI-WARE Consortium. "D.2.3.1b: FI-WARE Architecture." 2012-10-29. http://cordis.europa.eu/fp7/ict/netinnovation/deliverables/fi-ware/fi-ware-d231b.pdf (accessed 2017-03-17).

Gregor Hohpe, Bobby Woolf. *Enterprise Integration Patterns.* Boston, MA: Addison Wesley, 2011.

Jabeen, Hajira. *Big Data Europe Wiki.* 2016-07-26. https://github.com/big-data-europe/README/wiki (accessed 2017-03-17).

Pauwels, Erika. *Pipeline Builder Application.* 2016-09-06. https://github.com/big-data-europe/app-pipeline-builder (accessed 2017-03-27).

Platform, BDE. "BDE platform Wiki." 2016-03-31. https://github.com/big-data-europe/README/wiki/BDE-platform (accessed 2017-03-27).

Telefonica I+D, Fermin Galan (TID). *FIWARE.OpenSpecification.Data.ContextBroker.* Telecom Italia and Telefonica I+D. 2016. https://forge.fiware.org/plugins/mediawiki/wiki/fiware/index.php/FIWARE.OpenSpecification.Data.ContextBroker (accessed 2017-03-27).