Project funded by the European Union’s Horizon 2020 Research and Innovation Programme (2014 – 2020)

Support Action

Big Data Europe — Empowering Communities with Data Technologies

Project Number: 644564 Start Date of Project: 01/01/2015 Duration: 36 months

Deliverable 4.3, Final Big Data Integrator Platform Release

<table>
<thead>
<tr>
<th>Dissemination Level</th>
<th>Public</th>
</tr>
</thead>
<tbody>
<tr>
<td>Due Date of Deliverable</td>
<td>M35, 30.11.2017</td>
</tr>
<tr>
<td>Actual Submission Date</td>
<td>M36, 31.12.2017</td>
</tr>
<tr>
<td>Work Package</td>
<td>WP4, Big Data Integrator Platform &amp; Implementation</td>
</tr>
<tr>
<td>Task</td>
<td>T (4.1- 4.4)</td>
</tr>
<tr>
<td>Type</td>
<td>Other (Deployment Platform)</td>
</tr>
<tr>
<td>Approval Status</td>
<td>Final</td>
</tr>
<tr>
<td>Version</td>
<td>V1.0</td>
</tr>
<tr>
<td>Number of Pages</td>
<td>20</td>
</tr>
<tr>
<td>Filename</td>
<td>D4.3_Final Big Data Integrator Platform Release</td>
</tr>
</tbody>
</table>

Abstract:

This document presents a brief overview of the BDE platform released publicly on 16/11/2017.

The information in this document reflects only the author’s views and the European Community is not liable for any use that may be made of the information contained therein. The information in this document is provided “as is” without guarantee or warranty of any kind, express or implied, including but not limited to the fitness of the information for a particular purpose. The user thereof uses the information at his/ her sole risk and liability.
History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Reason</th>
<th>Revised by</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>16/11/2017</td>
<td>Initial version</td>
<td>Hajira Jabeen, (UBo)</td>
</tr>
<tr>
<td>0.1</td>
<td>29/11/2017</td>
<td>Final version</td>
<td>Hajira Jabeen, (UBo)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Jonathan Langens (TF)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Gezim Sejdiu (UBo)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Mohamed Nadjib Mami (FhG)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TenForce</td>
</tr>
<tr>
<td>1</td>
<td>21/12/2017</td>
<td>Review</td>
<td>Kiera Mcneice (OpenPhacts)</td>
</tr>
</tbody>
</table>

Author List

<table>
<thead>
<tr>
<th>Organization</th>
<th>Name</th>
<th>Contact Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>UBo</td>
<td>Hajira Jabeen</td>
<td><a href="mailto:jabeen@cs.uni-bonn.de">jabeen@cs.uni-bonn.de</a></td>
</tr>
<tr>
<td>InfAI</td>
<td>Ivan Ermilov</td>
<td><a href="mailto:iermilov@informatik.uni-leipzig.de">iermilov@informatik.uni-leipzig.de</a></td>
</tr>
<tr>
<td>UBo</td>
<td>Gezim Sejdiu</td>
<td><a href="mailto:sejdiu@cs.uni-bonn.de">sejdiu@cs.uni-bonn.de</a></td>
</tr>
<tr>
<td>Ten Force</td>
<td>Jonathan Langens</td>
<td><a href="mailto:jonathan.langens@tenforce.com">jonathan.langens@tenforce.com</a></td>
</tr>
<tr>
<td>FhG</td>
<td>Mohamed Nadjib Mami</td>
<td><a href="mailto:mami@cs.uni-bonn.de">mami@cs.uni-bonn.de</a></td>
</tr>
</tbody>
</table>
Table of Contents

1 Big Data Integrator Platform – 3rd Release 4
   1.1 Introduction 4
   1.2 Platform Overview 6
   1.3 Big Data Integrator (BDI) Integrated Development Environment (IDE) 7
      1.3.1 Architecture 8
      1.3.2 Docker 8
      1.3.3 Stack Builder 8
      1.3.4 Stack Editor 9
      1.3.5 Swarm UI 10
      1.3.6 BDE Logger 10
      1.3.7 Workflow Builder 11
      1.3.8 Init_daemon 12
      1.3.9 WorkFlow Monitor 13
   1.4 BDI Internal Architecture 13
   1.5 BDE ready tools 15
      1.5.1 Computational frameworks 15
      1.5.2 Data storage 15
      1.5.3 Data acquisition 15
      1.5.4 Search engines 16
      1.5.5 Distributed Key/Value Stores 16
      1.5.6 Semantic components 16
   1.6 User Instructions 16
2 Publications 17
3 BDE Adapters 18
4 Conclusion 19
5 References 20
1 Big Data Integrator Platform – 3rd Release

The Big Data Integrator (BDI) platform is designed to help communities in solving societal challenges and problems by accelerating the process of getting started with the Big Data technologies. With the 3rd release the BDE Tech team has completed following schedule of releases:

- December 2015,
- August 2016 and

The 3rd release was publicly announced on several channels including

- Twitter
- BDE Blog
- Conference EBDVF
- Press Release
- Mailing Lists

1.1 Introduction

The Big Data Europe platform enables developers to assemble big data repositories and/or streams and to process them in function of analysis and visualization. To this end the platform harnesses tools, workflows, pipelines and controls with an application for a set of selected pilots. The platform aspires to implement a generic, open and flexible architecture
such that it can accommodate for new, yet unforeseen tools and workflows within a transparent and easy to use environment. The early version of the Big Data Europe platform has been advanced substantially since D 4.2 (August, 2016) [2]. This third version includes a number of new features and extended existing ones in function of the ease of use and flexibility of the platform.

BDI platform has emerged as an easy-to-deploy, easy-to-use and adaptable (cluster-based and standalone) platform for the execution of Big Data frameworks and tools like e.g. Apache Hadoop, Apache Spark, Apache Flink and many others. BDI supports a wide range of tools reflecting the requirements gathered from the seven societal challenges. Thus, the platform allows execution of a variety of tasks like message passing (Kafka, Flume), storage (Hive, Cassandra), publishing (geotriples) and analytics (Spark, Flink). Overall, the platform has lowered the barrier to entry for new Big Data users and scientists from different domains to experiment with a variety of Big Data tools in a modular fashion.
1.2 Platform Overview

The detailed architecture of Big Data Integrator (BDI) is illustrated in Figure 1. BDI has made generous use of Docker ecosystem. Docker Swarm, with its built-in scheduler, offers features like scalability, interlinking of containers, networking among containers, resource management, load balancing, fault tolerance, failure recovery and log-based monitoring etc. The individual data processing applications of BDE are packaged as Docker images which makes sure that the applications will run as intended regardless of the host environment. Docker Compose helps in a simultaneous startup of multiple containers. The prerequisite of getting started with the BDI platform is the installation of Docker, Docker Compose and configuration of Docker Swarm. BDE has provided an easy-to-follow set of instructions and videos to install the platform in order to start working with the Big Data technologies.
The BDI has been built to ease the installation and development of Big Data tools. Therefore we have developed numerous additional services and features within Big Data Integrator - Integrated Development Environment (BDI-IDE). In the following sections, we will discuss the IDE.

1.3 Big Data Integrator (BDI) Integrated Development Environment (IDE)

BDI can be thought of as a "starter kit" for big data pipelines. A pipeline is the processing flow between collecting rough and large datasets and their aggregation into a data repository for analysis and visualisation. The BDI is the minimal standalone system providing a graphical user interface to a set of tools that are system independently wrapped in containers to help the users to create Big Data processing platforms.

---

**Figure 2: BDI workflow**
1.3.1 Architecture

BDI acts as a "skeleton" application where you can plug & play different Big Data services from the Big Data Europe platform.

At its core, it is a web application that renders different service's frontends in a single view, thus allowing the users to navigate between each service with a sense of workflow continuity (see Figure 2).

The initial startup of BDI-IDE provides several components to the users which are briefly covered below:

1.3.2 Docker

Basic building block for understanding and setting up a BDE is the Docker software layer that allows to configure and deploy applications inside a container. The container includes the entire environment to run a specific application.

A software application environment that consists of multiple applications or (micro-)services can be built with the 'Docker Compose' toolset. A Docker is defined by its Dockerfile that describes what is in the container and with what parameters? A Dockercomposefile (docker-compose.yml) describes which Dockers are required to build the intended software application whole.

1.3.3 Stack Builder

Stack Builder is a GUI to assemble and configure a new stack/pipeline. This can be done either by importing an already built definition from any git-repository or by creating an entirely new stack (no clone). The definition of a stack is a docker-compose.yml and it includes the description of the services to be deployed in the working environment. The components within the pipeline are then accessible for editing and specialized changes in the Stack Editor.
1.3.4 Stack Editor

The Stack Editor allows users to create a personalized `docker-compose.yml` by updating the imported file definition. It is equipped with suggestions & search features to ease the discovery and selection of components and configurations.
1.3.5 Swarm UI

After the `docker-compose.yml` has been created in the Stack Builder, it can be pushed into a git repository. From the SwarmUI, users can clone the repository and effectively launch an instance of (start, stop, restart, scale, etc.) the containers using docker swarm from within the same graphical user interface.

![Figure 4: Stack Editor](image)

1.3.6 BDE Logger

The architecture implies that all communication between containers goes through HTTP. The Logger service provides logging of all the HTTP traffic generated by the containers and pushes it into an Elasticsearch instance, where it can be visualized with Kibana. The Kibana tool can be configured with custom dashboards and data visualization to monitor a given instance of a BDE pipeline. Another use of this tool is the discovery of data. Kibana can be
used to search for and identify bottlenecks, failed calls, etc. It is also very easy to narrow down a system failure to the call responsible in a visual way.

The configuration necessary to enable HTTP logging for a certain microservice is done by adding a label "logging=true" to the labels of that microservice. The import of the current loggings into the Elasticsearch is done automatically. To import the data for visualization in Kibana the pattern has to be changed to "har*".

![Figure 6: Logger Visualization](image)

### 1.3.7 Workflow Builder

The Workflow Builder helps to define a specific set of steps that have to be executed in sequence, as a "workflow". Alternative name for ‘Workflow Builder’ is ‘Pipeline Builder’ while it allows to detail the specific sequence and configuration of each service to go from source data to results.
1.3.8 Init_daemon

To allow the Workflow Builder to enforce a workflow for a given stack (docker-compose.yml), the mu-init-daemon-service needs to be added as part of the stack. It will be the "referee" that imposes the steps defined in the workflow builder. "Init_daemon", given an application-specific workflow, orchestrates the initialization process of the components. It provides requests through which the components can report their initialization progress. The workflow builder reports the startup flow to init daemon that can validate whether a specific component can start based on the initialization status reported by the other components. The workflow needs to be described per application stack as it specifies the dependencies between services and indicates where human interaction is required. That service will be the "referee" that imposes the steps defined in the workflow builder. This adds functionality like Docker Healthchecks but more fine-grained.
1.3.9 WorkFlow Monitor

The Workflow Monitor is the Docker Swarm user interface. It allows a user to follow the initialization process. It displays the workflow as defined in the workflow builder application. For each step in the workflow, the corresponding status (not started, running or finished) is shown as retrieved from the init daemon service. The interface automatically updates when a status changes, due to an update through the init daemon service by one of the pipeline components. It also offers the option to the user to manually abort a step in the pipeline if necessary.

![Workflow Monitor](image)

**Figure 8: WorkFlow Monitor**

1.4 BDI Internal Architecture

The internal architecture of the BDI platform reuses also the microservice architecture allowing a maximum of flexibility and reusability through its modularity. The common
microservices to all architectures, like the identifier, dispatcher and the database will be featuring also in this platform.

Figure 9 clearly shows the constitution of front and backend. One can note that it is easy to add/remove/replace microservices and that the backend can also be accessed from another front end. The figure depicts the visual representation of the basic architecture.

![Figure 9: BDI Internal Microservices Setup](image-url)
1.5 BDE ready tools

Following tools have been provided with a container environment and a configuration so they can easily be reused in the BDI/BDE platforms. This list is not limited or exhaustive. With more practical deployments of the BDE application suite in different contexts more tools will receive a ‘wrapping’ to be reused easily in future data processing instances of BDE.

The latest status can be consulted at: https://github.com/big-data-europe/README/wiki/Components

1.5.1 Computational frameworks

- Flink
- Spark
- Storm

1.5.2 Data storage

- Hadoop
- Hue HDFS File Browser
- Cassandra
- HBase
- Hive
- Redis
- Virtuoso
- Zeppelin

1.5.3 Data acquisition

- Flume
- Message passing
1.5.4 Search engines

- Elasticsearch
- Solr

1.5.5 Distributed Key/Value Stores

- Zookeeper

1.5.6 Semantic components

- DEER
- EDCAT
- FOX
- GeoTriples
- Limes
- Silk
- SEMAGROW engine
- Sextant
- Strabon
- UnifiedViews

1.6 User Instructions

In order to facilitate the use of BDI and disseminate the work being done within the project, we have provided instructions and technical discussions on multiple channels. These include:

1. Blogs
2 Publications

There have been several publically appreciated (peer reviewed, international) research outcomes of the project.


7. “Simplifying the Deployment of Big Data Solutions” by Ivan Ermilov and Axel-Cyrille Ngonga Ngomo in KESW 2017 Demo/Poster Track [BibTex]

3 BDE Adapters

Below is the list of other softwares projects that are integrating BDE provided components into their systems.

1. **SANSA**
2. **proteus-h2020/proteus-docker** -Scalable online machine learning for predictive analytics.
3. **TSCache** uses Flink Docker for submitting jobs
4. **Joblib**
5. **Bigdata-docker**
6. **digital-assistance-system-cloud**
7. **onebox**
8. **caspervg/aggr**
9. **git-dev**
10. **project-ember**
11. **torus-docker-services**
4 Conclusion

The Big Data Integrator intends to ease the deployment and development of Big Data applications. It has been developed with the focus on two key aspects: ease of use and flexibility. With a consistent GUI, the user can select the tools that he needs for his business environment. He can define the workflow/pipeline and monitor the processing. The GUI lowers the threshold to make a new configured instance of the BDE platform adapted to the local needs and data sources.

To the best of our knowledge, BDI is the first open-source yet flexible and easy-to-use platform that allows the creation of a variety of workflows, application-stack alongside management and monitoring of the cluster status.

As being the first production ready version of the platform, we anticipate that the platform as a whole and the BDI in particular will benefit a lot from new features and enhancement to strengthen even more its robustness and application.
5 References
