

**EGI-Engage**

Deliverable/Milestone review form

|  |  |  |  |
| --- | --- | --- | --- |
| **Details of the document being reviewed** | | | |
| *Title:* | **Analysis on techniques to manage big data on the EGI accounting system** | *Document identifier:* | EGI-doc-2667 |
| *Project:* | **EGI-Engage** | *Document url:* | <https://documents.egi.eu/document/2667> |
| *Author(s):* | **Adrian Coveney**  **Stuart Pullinger** | *Date:* | **10-2-2016** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Identification of the reviewer** | | | |
| *Reviewer:* | **Christos Kanellopoulos** | *Activity:* | **JRA1** |

**General comments on the content**

|  |
| --- |
| **Comments from Reviewer:** |
| The structure of chapter 2 is confusing. The content is fine, but the organization of the content in current sections is not very meaningful and in one or two cases it is inaccurate. I propose that there is a section for **Datastores**. This can be split are split in SQL and NoSQL datastores. The latter can be further split in raw datastores, key value stores, document based stores, column based stores and graph based stores (although I do not see any graph based store mentioned here). Then you could have a section on **Tools for Data Ingestion** and then a section on **Computation/Query Engines.** In the latter you can mention RDBMS with parallel processing, stream and batch processing systems.  Something that was not mentioned in the document is the λ architecture, which is a very common pattern nowadays for batch and stream processing systems. Basically if you put together many of the tools you mentioned, you come up with a λ architecture. For your information, for the next generation of the ARGO Compute Engine we are going to use the λ architectural pattern and for the implementation a combination of Apache Kafka (internal message bus), Apache Flume (streams data to HDFS), Apache Hadoop/HDFS (underlying storage and batch compute layer), Apache Flink (Stream and Batch processing layer on top of Hadoop) and MongoDB for service layer (with a possible addition of Elasticsearch) |
| **Response from Author:** |
|  |

**Additional comments**

*(not affecting the document content e.g. recommendations for the future)*

|  |
| --- |
| **From reviewer:** |
|  |

**Detailed comments on the content**

| **N°** | **Page** | **§** | **Observations** | **Reply from author (correction / reject,  …)** |
| --- | --- | --- | --- | --- |
| 1 | 10 | 2.4 | Not all the tools mentioned in this section are built on top of HDFS |  |
| 2 | 11 | 2.4.1 | Consider changing the title “Database like systems”. Perhaps to (No-)SQL Database Systems. In general chapter 2 needs reorganization. |  |
| 3 | 11 | 2.4.1.2 | Pig is not a database system. It is a high level language that can be used to express complex computation that are then automatically compiled/translated to Hadoop Map/Reduce jobs |  |
| 4 | 11 | 2.4.2 | All of these are Stream processing frameworks, so I suggest that you change the title to that. One very prominent framework that is missing is “Apache Flink”. |  |
| 5 | 13 | 2.5 | I do not understand why there is a distinction between Apache and non-Apache tools. As I mentioned a couple of time already, I believe chapter 2 needs to be reorganized. |  |
| 6 | 22 | 5.2 | I do not understand why the summaries will be exported and republished to the MBN |  |

**English and other corrections:**

Note: English and typo corrections can be made directly in the document as comments.