BigStorage

PROPOSAL EVALUATION

Deliverable number	D1.3
Deliverable title	Final Report M48
	WP1 Use Cases
Editor	Toni Cortés (BSC)
Main Authors	All ESRs

Grant Agreement number	642963
Project ref. no	MSCA-ITN-2014-ETN-642963
Project acronym	BigStorage
Project full name	BigStorage: Storage-based convergence between HPC and Cloud to handle Big Data
Starting date (dur.)	1/1/2015 (48 months)
Ending date	31/12/2018
Project website	http://www.bigstorage-project.eu

Coordinator	María S. Pérez
Address	Campus de Montegancedo sn. 28660 Boadilla del Monte, Madrid, Spain
Reply to	mperez@fi.upm.es
Phone	+34- 910672857

Executive Summary

Although initially this was the document where all evaluation was to be presented, we have come to the conclusion that separating the technical proposal that is presented in the deliverables for WP2, WP3, WP4, and WP5 from their evaluation would make it difficult for readers to get a clear idea of the research outcome of the project. For this reason, we have decided to present these results in the deliverables of the WP where the technology has been discussed and, in this deliverable, we present a very short summary of such results pointing to the deliverables where the detailed information can be found (mainly D2.2, D3.2, D4.2,and D5.2).

Document Information

IST Project Number	MSCA-ITN-2014-ETN-642963
Acronym	BigStorage
Title	Storage-based convergence between HPC and Cloud to handle Big Data
Project URL	http://www.bigstorage-project.eu
Deliverable	D1.3 Final Report on WP1
Workpackage	WP1 Use Cases
Date of Delivery	Planned: 31.12.2016 Actual: 28.12.2016
Status	Version 1.0 final ■ draft □
Nature	prototype □ report ■ dissemination □
Dissemination level	public □ consortium ■
Distribution List	Consortium Partners
Responsible Editor	Toni Cortes (BSC), toni.cortes@bsc.es
Authors (Partner)	All ESRs
Reviewers	BigStorage Advisors
Abstract (for dissemination)	Executive Summary
Keywords	Performation evaluation, use cases

Version	Modification(s)	Date	Author(s)
0.1	Initial template and structure	28.12.2018	Toni Cortes, BSC
0.2	Final version	31.12.2018	Reviewed by Maria S. Perez, UPM

Project Consortium Information

Participants		Contact
Universidad Politécnica de Madrid (UPM), Spain		María S. Pérez Email: mperez@fi.upm.es
Barcelona Supercomputing Center (BSC), Spain		Toni Cortes Email: toni.cortes@bsc.es
Johannes Gutenberg University (JGU) Mainz, Germany		André Brinkmann Email: brinkman@uni-mainz.de
Inria, France		Gabriel Antoniu Email: gabriel.antoniu@inria.fr Adrian Lebre Email: adrien.lebre@inria.fr
Foundation for Research and Technology - Hellas (FORTH), Greece		Angelos Bilas Email: bilas@ics.forth.gr
Seagate, UK		Sai Narasimhamurthy Email:sai.narasimhamurthy@seagate.com
DKRZ, Germany		Thomas Ludwig Email: ludwig@dkrz.de
CA Technologies Development Spain (CA), Spain		Victor Muntes Email: Victor.Muntes@ca.com
CEA, France		Jacque Charles Lafoucriere Email: Charles.LAFOUCRIERE@CEA.FR
Fujitsu Technology Solutions GMBH, Germany		Sepp Stieger Email: sepp.stieger@ts.fujitsu.com

Table of Contents

1 Introduction

WP1 overview

The goals of this WP are to review in depth the technical and architectural needs of data storage for the 4 Use case's described (Smart Cities, HBP, SKA and Climate Science) to consolidate sets of requirements. These requirements have been fed into the various work packages. This WP iss also in charge of determining suitable benchmarks &specifications that reasonably represent these use cases and would be used to assess the BigStorage solutions derived by the different work packages. Delivery of these objectives has been organised to maximise the collaboration of ESR's across the project.

Structure of this document

Although initially this was the document where all evaluation was to be presented, we have come to the conclusion that separating the technical proposal that is presented in the deliverables for WP2, WP3, WP4, and WP5 from their evaluation would make it difficult for readers to get a clear idea of the research outcome of the project. For this reason, we have decided to present these results in the deliverables of the WP where the technology has been discussed and, in this deliverable, we will make a very short summary of such results pointing to the deliverables where the detailed information can be found.

2 Evaluations performed

WP2: Data Science

As ingestion for stream processing is a key problem in today's world, and as part of BigStorage, we have developed KerA, a novel ingestion system for scalable stream processing that introduces a dynamic partitioning scheme that elastically adapts to the number of producers and consumers by grouping records into fixed-sized segments at fine granularity and relies on a lightweight metadata management scheme that assigns minimal information to each segment rather than record, which greatly reduces the performance and space overhead of offset management, therefore optimizing sequential access to the records. We have performed a deep comparison with its main competitor Kafka, and how it reacts to the different configuration parameters.

Replication is key for both reliability and performance in in-memory key-value stores, and they induce a significant overhead. As part of BigStorage, we have developed Tailwind, a replication mechanism based on one-sided RDMA and we have extensively evaluated its overhead varying the environment and set of parameters.

Finally, the application of Artificial Intelligence techniques to Big Data architectures has been also evaluated, by means of Machine Learning-based tuning of Big Data architectures parallelization settings of Spark and Flink and the use of A* and Hill Climbing techniques to graphs, representing faults in microservice architectures.

The detailed results are presented in D2.2.

WP3: HPC-Cloud convergence

Several works have shown that the time to boot one virtual machine (VM) can last up to a few minutes in high consolidated cloud scenarios. This time is critical as VM boot duration defines how an application can react w.r.t. demands' fluctuations (horizontal elasticity). To limit as much as possible the time to boot a VM, we design the YOLO mechanism (You Only Load Once), a new storage service that uses HPC caching approaches to speed VM Boot Time duration. We have evaluated it and showed a significant reduction in the booting time of VMs when using our proposal.

HPC application are starting to use new data-intensive access patterns, and we have developed GekkoFS to address them. The file system provides relaxed POSIX semantics, only offering features which are actually required by most (not all) applications. It is able to provide scalable I/O performance and reaches millions of metadata operations already for a small number of nodes, significantly outperforming the capabilities of general-purpose parallel file systems.

Týr takes the benefit of the growing trend of equipping HPC compute nodes with local storage in order to deploy object storage systems alongside the application on the compute nodes. By leveraging Blobs

(Binary Large Objects) as an alternative to files, Tyr improves storage throughput for a variety of existing workloads

The detailed results are presented in in D3.2

WP4: Storage solutions

As part of WP4, we have developed and evaluated several techniques. In this section we will sketch all of them from the ones closer to the HW to the ones closer to the user.

Failures in data centers produce significant economical loses, thus predicting failures to replace components has become very important. We have performed a cross-system root cause classification framework based on similarity evaluation of weighted graphs with multi-attribute nodes to predict such failures and we have evaluated it confirm the quality of root cause classification.

One of the problems NAND storage has is the overhead introduced by the garbage collector and especially when redundancy is used to increase reliability. Thus, we have analysed the write amplification when using the RAIN model and the benefits we observed when the GC collector was coordinated.

In this WP, we have worked on the definition of prefetching algorithm based on static code analysis instead of the traditional past execution behaviour. This has been evaluated over a next-generation object store (dataClay) and on a triple store (DBpedia).

Files are traditionally thought as an indivisible piece of data and is stored in a single device (at most stripped over several identical devices). We have evaluated the performance benefits of arbitrarily dividing a file over the most adequate storage components taking into account the different parts of a file (index, most used blocks …).

In a similar way, we have proposed a composable system that enables system administrators to place/replicate/stripe date over different storage systems (HOC, cloud, local…). We have compared this mechanism with GlusterFS, a well-established storage system powered by RedHat, obtaining significant performance improvements in addition to greater flexibility.

Adequate data parallelization is key to achieve good performance. Thus, we have proposed a method to recommend optimal parallelization settings to users depending on the type of application. We solve this optimization problem through machine learning, based on system and application metrics collected from previous executions. We have evaluated these recommendations and have shown that they are near optimal.

The detailed results are presented in D4.2 and D2.2 (for part of the root cause analysis evaluation).

WP5: Energy consumption

As part of WP5, we have evaluated the energy consumption of in memory key-value stores, mainly in the replication mechanisms. This evaluation has been performed using RAMCloud as an example but has been generalized of this kind of in-memory object stores and has been divided into the different sources (CPU, networking, disk…).

We have also evaluated the effect of the data reduction in the energy consumption for data stored in HDF5 and netCDF formats. In this evaluation, different compression algorithms and datasets have been used.

The detailed results are presented in D5.2.