Accepted Manuscript

BDWatchdog: Real-time monitoring and profiling of Big Data applications and frameworks

Jonatan Enes, Roberto R. Expósito, Juan Touriño

PII: S0167-739X(17)31609-6
DOI: https://doi.org/10.1016/j.future.2017.12.068
Reference: FUTURE 3906

To appear in: *Future Generation Computer Systems*

Received date : 21 July 2017
Revised date : 5 October 2017
Accepted date : 31 December 2017

Please cite this article as: J. Enes, R.R. Expósito, J. Touriño, BDWatchdog: Real-time monitoring and profiling of Big Data applications and frameworks, *Future Generation Computer Systems* (2018), https://doi.org/10.1016/j.future.2017.12.068

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

ACCEPTED MANUSCRIPT

BDWatchdog: real-time monitoring and profiling of Big Data applications and frameworks

Jonatan Enes*, Roberto R. Expósito, Juan Touriño

Computer Architecture Group, Universidade da Coruña, Campus de A Coruña, 15701 A Coruña, Spain

Abstract

Current Big Data applications are characterized by a heavy use of system resources (e.g., CPU, disk) generally distributed across a cluster. To effectively improve their performance there is a critical need for an accurate analysis of both Big Data workloads and frameworks. This means to fully understand how the system resources are being used in order to identify potential bottlenecks, from resource to code bottlenecks. This paper presents BDWatchdog, a novel framework that allows real-time and scalable analysis of Big Data applications by combining time series for resource monitorization and flame graphs for code profiling, focusing on the processes that make up the workload rather than the underlying instances on which they are executed. This shift from the traditional system-based monitorization to a process-based analysis is interesting for new paradigms such as software containers or serverless computing, where the focus is put on applications and not on instances. BDWatchdog has been evaluated on a Big Data cloud-based service deployed at the CESGA supercomputing center. The experimental results show that a process-based analysis allows for a more effective visualization and overall improves the understanding of Big Data workloads. BDWatchdog is publicly available at `http://bdwatchdog.dec.udc.es`.

Keywords: Big Data, monitoring, profiling, time series, flame graphs, process-based analysis

1. Introduction

In recent years, the use of Big Data frameworks and their resource demands are increasing at an astounding rate, both from the business world as well as from the research front. With the appearance of distributed processing frameworks such as Apache Hadoop [1] and subsequent evolutions like Apache Spark [2], Big Data workloads have evolved and adapted to the users' needs and use cases as well as to the underlying hardware infrastructure that supports them. Nevertheless, it is increasingly well established that Big Data workloads are characterized by a heavy use of different system resources (i.e., CPU, memory, disk and network) in one way or another, and thus in order to increase their efficiency several improvements are possible.

Much research has already been conducted on analyzing Big Data workloads and frameworks in order to either minimize their resource consumption and optimize them as a whole [3, 4, 5, 6] or to increase infrastructure consolidation and efficiency if cloud computing or hypervisor-based virtualization technologies are used [7, 8]. However, when it comes to a more fine grain level of system monitorization, there are no widespread solutions capable of targeting isolated processes (i.e., being a process an instance of a running application on an operating system), and in the end this is disregarded in favor of overall instance resource usage. Nevertheless, this step forward is especially important when considering emerging technologies like operating-system-level virtualization based on software containers (e.g., Docker [9]) or new compelling cloud-based models like serverless computing [10]. Both technologies use environments where applications are no

*Corresponding author. Tel.: +34 881 011 212; Fax: +34 981 167 160

Email addresses: `jonatan.enes@udc.es` (Jonatan Enes), `rreye@udc.es` (Roberto R. Expósito), `juan@udc.es` (Juan Touriño)

Download English Version:

https://daneshyari.com/en/article/6872933

Download Persian Version:

https://daneshyari.com/article/6872933

Daneshyari.com