2019 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)

Applying Combinatorial Testing to Large-scale Data Processing at Adobe

Riley Smith
Adobe Inc.
Utah, USA
rilsmith@adobe.com

Darryl Jarman
Adobe Inc.
Utah, USA
djarman@adobe.com

Richard Kuhn
NIST
Maryland, USA
d.kuhn@nist.gov

Raghu Kacker
NIST
Maryland, USA
raghu.kacker@nist.gov

Dimitris Simos
SBA Research
Vienna, Austira
dsimos@sba-research.org

Ludwig Kampel
SBA Research
Vienna, Austria
lkampel@sba-research.org

Manuel Leithner
SBA Research
Vienna, Austria
mleithner@sba-research.org

Gabe Gosney
Adobe Inc.
Utah, USA
gosney@adobe.com

***Abstract*- Adobe offers an analytics product as part of the Marketing Cloud software with which customers can track many details about users across various digital platforms. For the most part, customers define the amount and type of data to track. This high dimensionality makes validation difficult or intractable. Due to increasing attention from both industry and academia, combinatorial testing was investigated and applied to improve existing validation. In this paper, we report the practical application of combinatorial testing to the data collection, compression and processing components of the Adobe analytics product. Consequently, the effectiveness of combinatorial testing for this application is measured in terms of new defects found rather than detecting known defects from previous versions. The results of the application show that combinatorial testing is an effective way to improve validation for these components of Adobe Analytics. In addition, we report the details of the input parameter modeling process and test value selection to provide more context for the problem and how combinatorial testing provides the structure to improve validation for Adobe Analytics.**

Keywords- Combinatorial Testing, Industry, Application

I. INTRODUCTION

Originating from web analytics, the Adobe analytics product has evolved into a customer marketing platform allowing users to instrument data collection across many digital platforms for real-time reporting. Users of Adobe Analytics configure the amount and type of data to track. The available configurations result in high dimensionality for any elements of the system that interact with the collected data. For example, the collected data are the main input parameters for the data collection, compression, and processing components. As the product has evolved, the number of configurable elements has increased to at least a few thousand just for these components. Given this domain knowledge, traditional validation of these components relied on randomly generated values for the data input parameters. This approach was generally seen as a practical solution to exercise the input space based on the assumption that the input space was too broad to systematically cover. Over time, faults in these components exposed interactions not covered by this traditional approach. These faults revealed the insufficiency of this existing validation method.

A key observation of combinatorial testing maintains that software faults are generally caused by the interactions between a limited (small) number of input parameters [1]. Generally, a t-way combinatorial test covers all t-way interactions. After discovering combinatorial testing, initial investigations revealed several reports showing the effectiveness in practical industry applications [2]. Despite many being labeled as “uncontrolled” applications and studies [3], these reports prompted the internal tools team at Adobe Analytics to apply combinatorial testing to provide better values for the data collection input parameters.

In this paper, we consequently report an industry application of combinatorial testing to the data collection, compression, and processing components of Adobe Analytics. Intending to improve existing validation, the effectiveness of combinatorial testing is measured in terms of new faults found rather than detecting known defects in previous faulty versions. Initial results of this combinatorial testing application found new faults in each of the subject systems with a small set of test cases. For example, a significant fault was detected in the data compression algorithm by a 2-way test set containing only ~150 tests. Furthermore, the results suggest that combinatorial testing may prove more effective than the traditional random approach.

It is important to note that the subject systems vary in size in terms of lines of code, but all have a large number of input parameters with complex constraints and many possible values. This has two main implications that affect implementation of combinatorial testing: (1) no tool existed for creating covering arrays that supported so many input parameters and (2) the values for the input parameters needed to be minimized.

978-1-7281-0888-9/19/$31.00 ©2019 IEEE
DOI 10.1109/ICSTW.2019.00051