Think Outside the Box: Exploring and Exploiting Cross-Language Optimizations in MapReduce

Marija Selakovic TU Darmstadt Germany m.selakovic89@gmail.com

Madan Musuvathi Microsoft Research USA madanm@microsoft.com Michael Barnett Microsoft Research USA Michael.Barnett@microsoft.com

> Todd Mytkowicz Microsoft Research USA toddm@microsoft.com

ABSTRACT

ACM Reference Format:

1 INTRODUCTION

RQ:

- What is the proportion of time spent in native vs. non-native job vertices?
- What are the root causes of running job vertices as non-native?
- How much of the non-native vertices can be translated to native?

2 BACKGROUND

2.1 Scope Language

2.1.1 Execution of a Scope Job.

- 2.2 Running C++ vs. C# Vertex
- 3 METHODOLOGY
- 3.1 Subject Programs and Artifacts
- 3.2 Scale of The Study
- 4 RQ1: NATIVE VS. NON-NATIVE TIME
- 5 RQ2: ROOT CAUSES OF NON-NATIVE VERTICES
 - .NET Framework Calls
 - User written functions
 - Custom processors, reducers, combiners, extractors, etc.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Conference'17, July 2017, Washington, DC, USA
© 2017 Copyright held by the owner/author(s).
ACM ISBN 978-x-xxxx-xxxx-xYY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnn

6 RQ3:TRANSLATION OF C# TO C++

7 CASE STUDIES

description for each optimized job if possible

- 7.1 Case Study 1
- 7.2 Case Study 2

8 LIMITATIONS AND THREADS TO VALIDITY

Underapproximation of performance impact.

Granularity of the analysis.

Challenges for implementing more intrinsics.

9 RELATED WORK

9.1 Query Optimizations

•••

10 CONCLUSIONS REFERENCES