

CS 461 — Fall 2016 — Problem Statement

Leon Leighton, Thomas Olson, Derek Wong

POWER8 Build and Test Infrastructure

Abstract

Publicly available Continuous Integration systems provide a means for Open Source software developers to validate their software after each change to the code. While such systems exist for other architectures, there is currently no public Continuous Integration system available for IBM's POWER8 architecture. Correcting this absence will help to ensure the existence of a healthy, diverse, architecture environment that enables flexibility in finding problem solutions. Users of this project will have an easy to use method to ensure that their software works correctly on POWER8 systems running the Linux operating system.

PROBLEM DEFINITION

Open Source software projects are often designed to work on as broad a range of hardware architectures as possible. The users of Open Source software want the flexibility to consider a variety of hardware solutions in order to take advantage of different capabilities and to avoid vendor lock-in in the future. To support their users, Open Source software projects spend their time and resources on adding support for new architectures as they become available. The introduction of IBM's POWER8 architecture broadened the range of possible solutions available to the end users of Open Source software. This creates a need for Open Source projects to ensure that their software builds and functions correctly on POWER8. Open Source software projects often use Continuous Integration (CI) systems to build and test their software. CI enables these projects to validate their software after each change, ensuring that it continues to work as expected. Public Cloud-based solutions already exist to enable Open Source software developers to build and test their software on Intel's x86 architecture. Thus, while much of the work needed to enable software to compile on POWER8 has already been done at the compiler level, Open Source software developers still need an easy way to build and test their software on POWER8.

PROPOSED SOLUTION

This project will focus on providing an easy to use, public, Cloud-based, POWER8 CI infrastructure for Open Source projects in order to increase the ability of these projects to build and test their software on POWER8. Creating this POWER8 CI system will lower the cost and effort needed for Open Source software developers to ensure that their software runs on POWER8. The Oregon State University Open Source Lab (OSL) currently has a POWER8 cluster running OpenStack. We will use this POWER8 Cloud along with the Jenkins automation software to create a POWER8 CI system for Open Source projects that will enable automatic validation after every change to the software. This system will be able to use both Linux virtual machines and Docker containers to build the software and run associated tests. We will also focus on ease of use through simple configuration and by allowing developers to use GitHub, and possibly other, credentials for authentication.

Another area of concern will be security. In addition to the aforementioned authentication, we will ensure that the system is designed in such a way that project builds will be isolated from each other. We want to avoid the possibility of any project build being able to compromise the availability of the system to other projects or to compromise the integrity of binaries that are built by the system.

The project itself will also be Open Source which will allow others to replicate our solution on Clouds other than the one we will be utilizing at the OSL. This will be accomplished by hosting the project on GitHub.

During the Engineering Expo in the Spring we will be able to show various Open Source projects being built and tested on POWER8 using this infrastructure.

PERFORMANCE METRICS

We will measure the success of our project by having a functioning POWER8 Continuous Integration infrastructure that is easy to use by Open Source projects.

APPROVAL

Gerrit Huizenga _____

Leon Leighton _____

Thomas Olson _____

Derek Wong _____