POWER8 Continuous Integration Design Document

Leon Leighton, Thomas Olson, Derek Wong
Project 35

Abstract

This document contains the initial design of the POWER8 CI project in the form of user stories. Each user story contains a requirement, how we intend to meet that requirement, and how the meeting of the requirement can be confirmed.

Approval

Gerrit Huizenga ____________________

Leon Leighton ____________________

Thomas Olson ____________________

Derek Wong ____________________

CONTENTS

I. INTRODUCTION

The goal of this document is to communicate the requirements and design of the POWER8 Continuous Integration project. This information is conveyed in the form of user stories which contain three parts. The 'CARD' contains the requirement and the role of the stakeholder. The 'CONVERSATION' conveys how we intend to meet that requirement. Finally, the 'CONFIRMATION' indicates a way that the meeting of the requirement can be verified. As can be seen from the user stories, the overall design of this project is one of integrating a number of existing resources and open source projects into a system which will provide a continuous integration system for IBM's POWER8 hardware.

Below is a list of the user stories and the author.

POWER8 Cluster POWER8 servers that can be used for Continuous Integration. Authored by Leon Leighton.

Deployment/Configuration Management Manage installing the POWER8 CI system in a reproducible way. Authored by Leon Leighton.

Isolation Isolate builds from each other and other users of the cluster. Authored by Leon Leighton.

Resource Management Manage resources by automating the creation and removal of virtual machines and containers. Authored by Leon Leighton.

Login Use an existing login. Authored by Derek Wong.

Build Status Informing user of build status. Authored by Derek Wong.

Persistant VMs Persistent virtual machines for our website and Jenkins. Authored by Derek Wong.

Build Artifacts Access the built and tested files. Authored by Thomas Olson.

Build/Environment Configuration Configure the build and testing environment and methods. Authored by Thomas Olson.

Automation of Builds Automatically run builds when a commit is made to the repository. Authored by Thomas Olson.

Pre-build VM images The system should use pre-built images for the base system of VMs. Authored by Thomas Olson.

II. USER STORIES

A. *POWER8 Cluster*

1) CARD: As an open source software developer, I want to build and test my software on POWER8, so that I can ensure proper functioning on that architecture.

2) CONVERSATION: The goal of this project is to enable open source software projects to build and test their software on POWER8. The OSU Open Source Lab has a POWER8 cluster that will serve as the hardware basis for this project. The OSL POWER8 cluster runs OpenStack [1] and is managed via the Chef [2] configuration management system. We will be working closely with the OSL to ensure that we have access to the POWER8 cluster for both testing and deployment of our Continuous Integration system. We will also work with the OSL to ensure that any needed OpenStack components are installed.

3) CONFIRMATION: The POWER8 CI team members, in coordination with the OSL, are able to deploy the CI system on the OSL's POWER8 OpenStack cluster.

B. *Deployment/Configuration Management*

1) CARD: As a system administrator, I want to centralize the management of the POWER8 CI system, so that there is a single source of truth for the configuration of the system.

2) CONVERSATION: We will need to deploy and configure multiple components to create the POWER8 CI system. In addition, one of the goals of our project is to enable others to recreate our POWER8 CI system. We can meet these needs and goals by using a configuration management system. Ansible [3] is a IT automation solution that will allow us to manage the configuration and deployment of our CI system. Specifically, we will be using Ansible playbooks to install and configure Jenkins and Jenkins plugins and any additional components of the CI system. Also, we will be keeping the Ansible playbooks in version control to allow us to keep track of changes and rollback changes when necessary. This will give us the flexibility we need to test changes while being able to return to a known good configuration.

3) CONFIRMATION: Ansible playbooks exists that will deploy our CI system on a POWER8 OpenStack cluster.

C. Isolation

1) CARD: As an open source software developer, I want my project builds to be isolated from other projects, so that my builds and test complete without outside interference.

2) CONVERSATION: Our project will be hosted on the OSL's POWER8 OpenStack cluster and we want to ensure that our usage of the cluster does not negatively impact other users of the system. Additionally, we do not want to run builds and tests in such a way that our users could interfere with each other. We can achieve these ends by isolating builds and tests in virtual machines or Linux containers. We will work with the OSL to determine appropriate quotas for usage of CPU, memory, and storage resources. Our primary means of launching and removing virtual machines and containers will be through OpenStack's Nova and Magnum components. By using these components we can leverage OpenStack's ability to manage virtual machines and Linux containers to provide isolation between different projects' builds and tests.

3) CONFIRMATION: Projects are built in virtual machines or containers.

D. Resource Management

1) CARD: As an open source software developer, I want my project builds to finish in a timely manner, so that I can receive build and test feedback as quickly as possible.

2) CONVERSATION: The users of our project want their builds to run as soon as possible. However, our project has a finite amount of resources available on the POWER8 OpenStack cluster. This may cause a delay in starting a build if the necessary resources are not available. Builds should be queued to run in the order received if the resources are not available to run them immediately. In order to ensure that resources are available for new builds, we will need to automatically remove old VMs and containers. As mentioned in another user story, OpenStack has the ability to manage virtual machines and containers through the Nova and Magnum components. In order to automate this, we will utilize Jenkins [4] which has the ability to interact with OpenStack by using a plugin [5]. This will give us the ability to automate both the creation and removal of VMs and containers.

3) CONFIRMATION: Builds are triggered immediately if resources are available, and queued in the order recieved otherwise.

E. Login

1) CARD: As an open source software developer, I want to be able to use my github account credentials to login, so that I don't have to create another account, to access Jenkins.

2) CONVERSATION: We will use pluggable authentication from Jenkins to accomplish this task. Specifically, we will use the GitHub OAuth Plugin to allow users to login with their GitHub credentials.

3) CONFIRMATION: Our registration web site will allow developers to simply use their GitHub credentials to login into our services.

F. Build Status

1) CARD: As an open source software developer, I want to see my build status, so I know if my build passed or failed.

2) CONVERSATION: We will use a plugin from Jenkins to display the status of our developer's build. More specifically, we intend to use the Build Monitor Plugin [6] to display the information to our users. The Build Monitor Plugin supports many features such as: display the status and progress of selected jobs, display the names of people who might be responsible for 'breaking the build', display who is fixing a broken build, and display what broke the build. The plugin can be installed through Jenkins.

3) CONFIRMATION: Our web site will allow developers to see their build status (pass/fail).

G. Persistent VMs

1) CARD: As a system administrator, I want persistent virtual machines (VMs) for our website and Jenkins, so that I have a platform to host my service.

2) CONVERSATION: We will use OpenStack to deploy our virtual machines. Then we will use those virtual machines to host our website and Jenkins. OpenStack is a set of software tools for building and managing cloud computing platforms, more specifically it is used to deploy virtual machines to handle different tasks. For our website, we will be using Apache HTTP Server.

3) CONFIRMATION: Our website and Jenkins will be hosted on virtual machines.

H. Build Artifacts

1) CARD: As an open source software developer, I want to access the built and tested files, so that I can distribute them to users.

2) CONVERSATION: Developers need to be able to access the created binaries and other files from building the source code, so that they may be able to distribute them as needed on their own.

3) CONFIRMATION: The files can be downloaded from the website.

I. Build/Environment Configuration

1) CARD: As an open source software developer, I want to configure the build and testing environment and methods, so that I can be sure it matches my standard build process.

2) CONVERSATION: Each repository needs a way to configure things such as required packages for building and command-line arguments. To that end, the system will require a configuration file in YAML format. This file will contain information such as necessary libraries and packages, programs and command-line arguments for building, and how to test the built code.

3) CONFIRMATION: The system checks for a file called '.powerci.yml' in registered repositories and configures the build environment with it.

J. Automation of Builds

1) CARD: As an open source software developer, I want the system to automatically run builds when a commit is made to the repository, so that I can determine if a bug has been introduced as early as possible.

2) CONVERSATION: Developers require a system that will automatically pull, build and test so they can always have the most current binaries available and so they can quickly know if a recent change causes the build to fail. Jenkins has plugin support for polling Git repositories for changes and then executing a pull-build-test-deploy sequence when a commit is detected to have been pushed since the last build was ran.

3) CONFIRMATION: The system automatically executes a pull-build-test-deploy sequence when a commit is pushed to the Git repository.

K. Pre-build VM images

1) CARD: As an open source software developer, I want to be able to choose from pre-built virtual machine, so that my builds and tests can target a known platform.

2) CONVERSATION: Pre-built images that are used for spinning up VMs makes the entire process of deploying and configuring a VM for running a pull-build-test sequence much faster and smoother. Ubuntu will probably be the OS of choice for the base images, however we may look into providing images for other Linux distributions.

3) CONFIRMATION: The system uses disk images with Linux installed for starting VMs.

References

[1] OpenStack Foundation. Openstack open source cloud computing software. [Online]. Available: http://www.openstack.org/
[2] Chef Software. Chef – embrace devops. [Online]. Available: https://www.chef.io/
[3] Red Hat. Ansible is simple it automation. [Online]. Available: https://www.ansible.com/
[4] Jenkins Project. Jenkins. [Online]. Available: https://jenkins.io/
[5] ——. Jclouds plugin – jenkins – jenkins wiki. [Online]. Available: https://wiki.jenkins-ci.org/display/JENKINS/JClouds+Plugin
[6] J. Molak. (2016, Nov) Build monitor plugin. [Online]. Available: https://wiki.jenkins-ci.org/display/JENKINS/Build+Monitor+Plugin