OpenStack at NDSU - Design Document

Justin Anderson
John Nagel
Cesar Ramirez
John Rensberger

Table of Contents

1. Overview

1.1 Introduction

OpenStack is an open-source software project for deploying private and public clouds on commodity hardware. Big companies like Amazon, Microsoft and Google have big 'clouds' running on high-end server-quality computers. Until recently, it was impossible for a small group of people to deploy a private cloud due to the prohibitive cost of hardware, software, and maintenance. OpenStack provides a solution to this problem by using existing services and interconnecting them with an API driven interface.

OpenStack was started as a collaboration of NASA and RackSpace in July of 2010. Since then, the project has received a lot of support from the open source community as well as big technologies companies like AMD, Intel, Canonical, SUSE Linux, Red Hat, Cisco, Dell, HP, IBM, NEC, VMware, and Yahoo!.

IBM's dedication in OpenStack is very clear, as it has been a Platinum Sponsor of the OpenStack Foundation since 2012. IBM has staff dedicated to improving OpenStack and is also working to promote the use of OpenStack in environments where it was previously not feasible such as education. Private clouds provide resources in a powerful and flexible capacity and can be efficiently utilized and advanced in academia.

1.2 Scope

The scope for this project is going to be to establish an OpenStack infrastructure that will allow professors, researchers and students to run programs on a private cloud hosted by the computer science department. We will focus on obtaining requirements from professors and researchers, analyzing their requests and selecting those that are most appropriate, and finally we will implement as many as possible. We will also work with faculty to explore the feasibility of migrating existing research projects to the OpenStack platform.

2. High-Level Design

OpenStack is composed of a set of services that consist of various programs. These services complement one another to achieve common goals to produce a cloud environment. At the highest level, the current OpenStack release is composed of seven services: Nova (compute), Swift (object storage), Glance (image service), Keystone (identity), Horizon (dashboard), Quantum (networking), and Cinder (block storage). All these services are capable of running on different kinds of hardware, from ARM clusters to commodity x86 hardware to enterprise-level servers.

2.1 High-Level Component Design

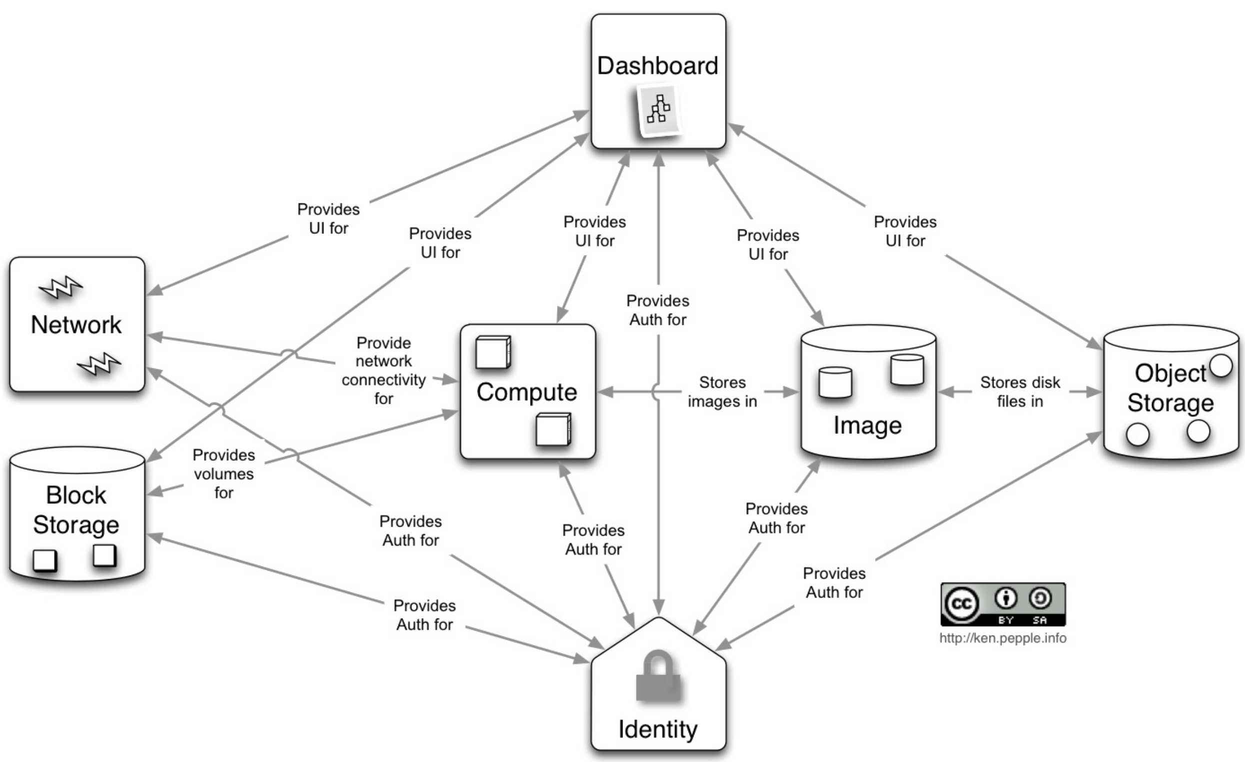
Component	Description
Nova - Compute	The Nova service handles the load of the virtualization across multiple architectures and backends. Some backends are Hyper-V, KVM, QEMU and XenServer. Nova can also run on certain ARM chips.
Swift - Object Storage	The Swift service is in charge of storing and retrieving information stored across multiple nodes. By using this distributed architecture it is by-design that inexpensive hard drives and servers can be used while still providing high availability. Swift will check for data redundancy and integrity.
Glance - Image Service	Glance provides a repository of images that can be retrieved in order to create new VMs.
Keystone - Identity	The Keystone service is in charge of authenticating the users and services that interact with OpenStack. Keystone also supports multiple backends, such as LDAP and other SQL based databases.
Horizon - Dashboard	Horizon is a Python-based front-end that allows administrators and end-users to interact with OpenStack services.
Quantum - Networking	Quantum is a service that provides an API driven system for managing networking within the cloud.
Cinder - Block Storage	Cinder is the service that handles persistent block level storage for the VMs.

2.2 Required Services

Component	Related Requirements
All	Gigabit wired networking between all components
Nova	Hardware capable of virtualization
Swift	Redundant and distributed storage
Glance	Storage for all images
Keystone	Authentication backend
Horizon	Python enabled web service
Quantum	High reliability network infrastructure
Cinder	Sufficient storage for VMs and snapshots

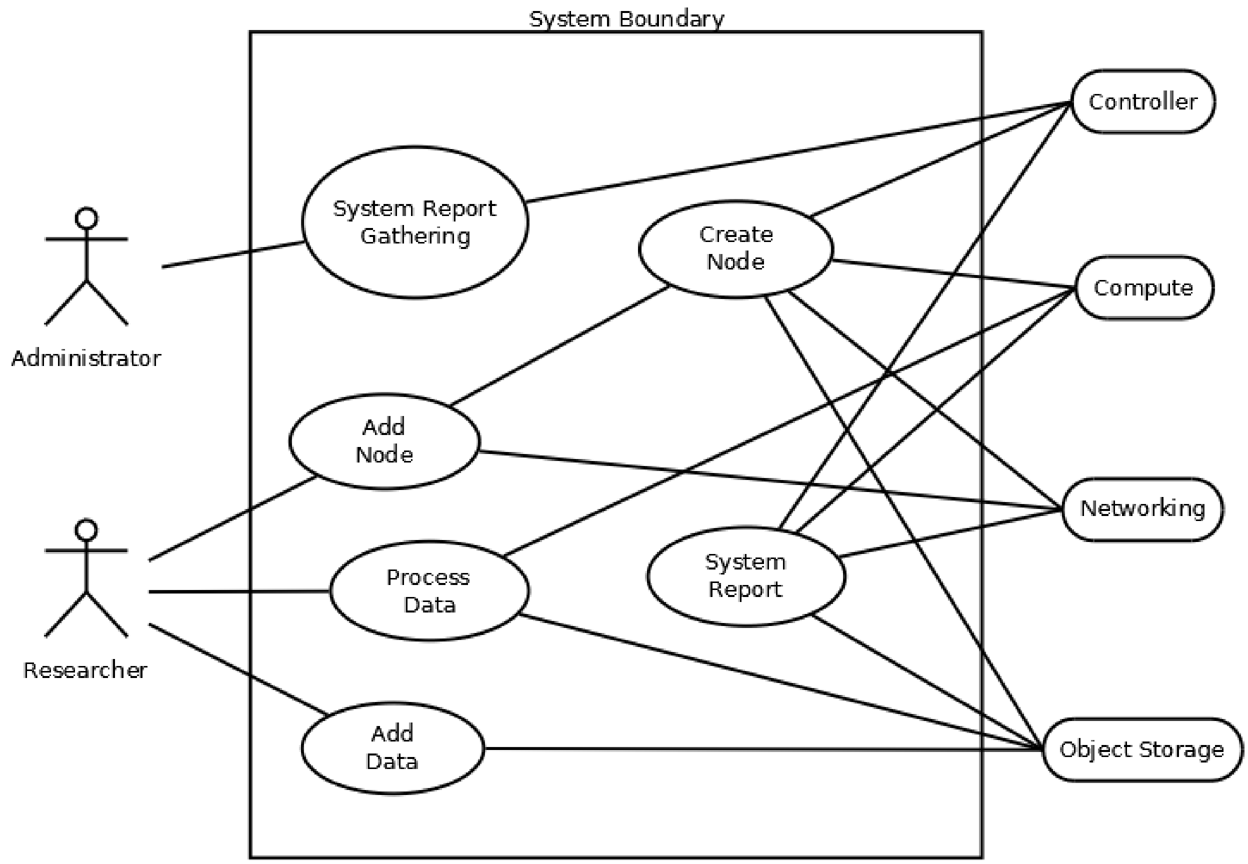
2.3 Activity Diagram

The following is a diagram of the current architecture of OpenStack.



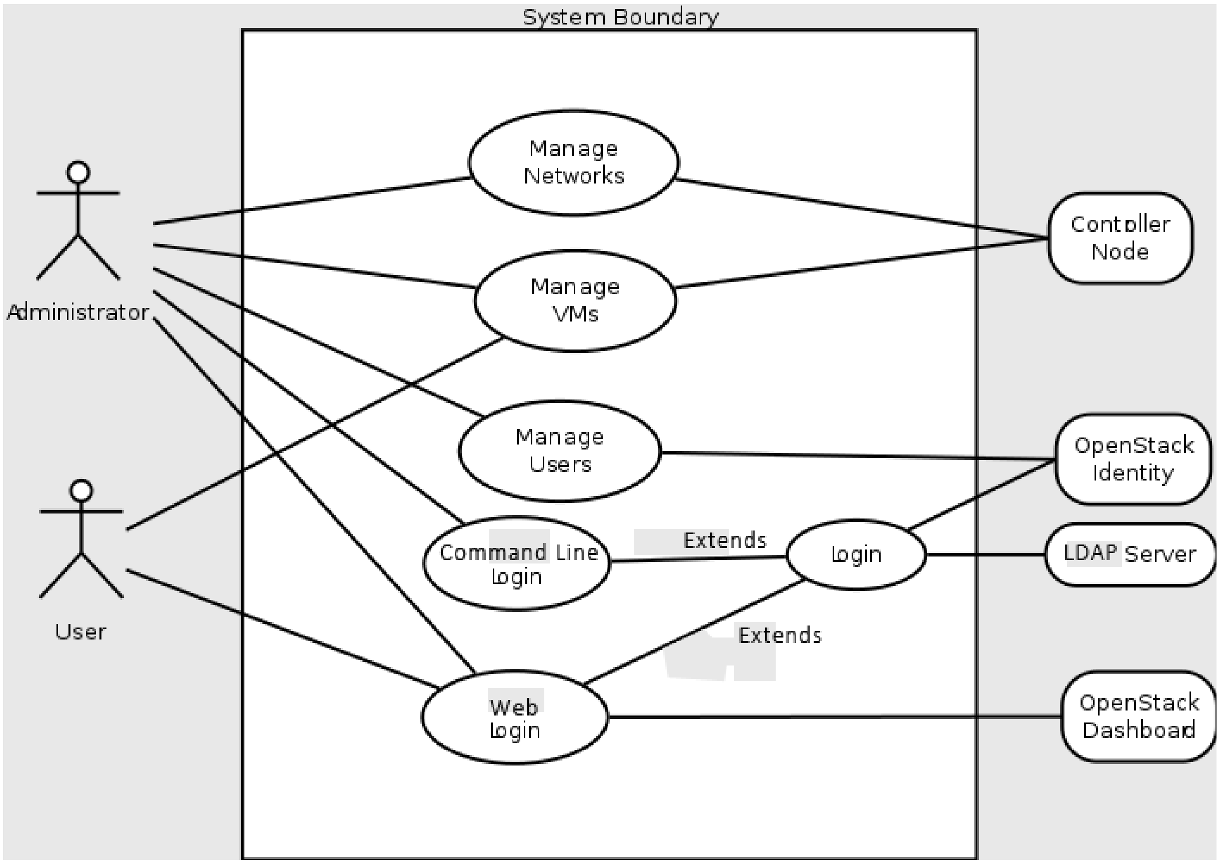
2.4 Use Case Diagrams

Use Case 1: Storing and processing large amounts of farming data.



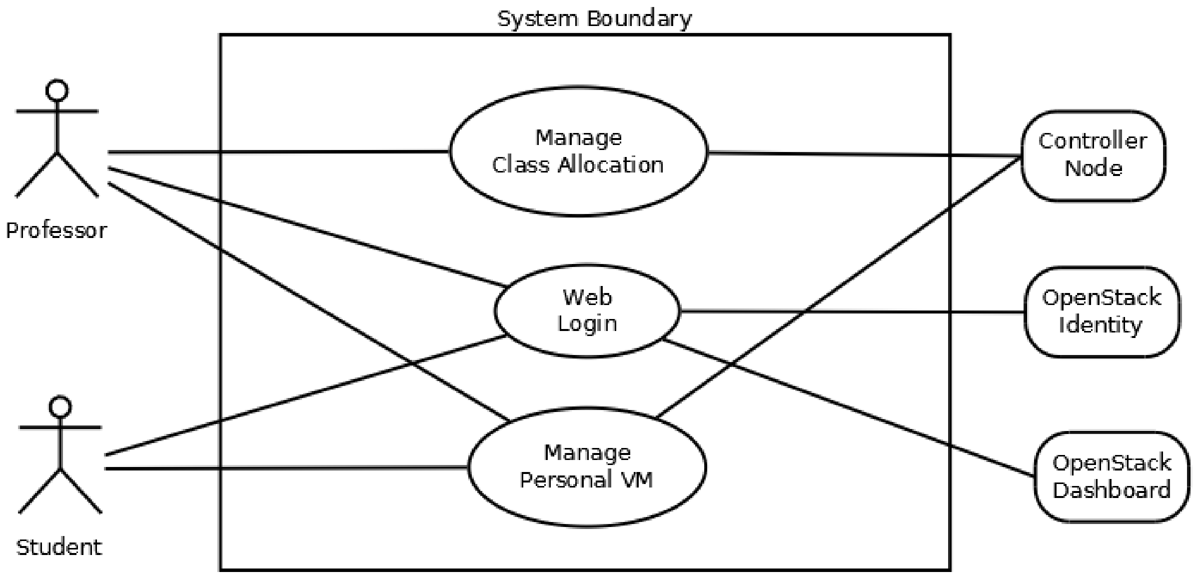
There is a project led by Dr. Anne Denton and Dr. Simone Ludwig that requires a place to store large amounts of data in a distributed file system. This data will later be processed using Hadoop. This project requires us to allow researchers to create Hadoop nodes on demand and combine them into a Hadoop cluster. This project will be receiving data from farmers in a weekly basis, and the researchers should be able to add this data as objects into the object storage. As an extra requirement, the system administrator should be able to receive reports on all OpenStack components to determine their working status.

Use Case 2: Administrators will be able to allocate resources to authenticated users.



Administrator will provide users with resources such that they can create and destroy their own VMs. Both administrators and users will need to authenticate before doing any action. Users will use the web interface to login and manage their allocation. In the case of an administrator, they should be able to login via the web interface as well as the command line interface. Both login interfaces will query OpenStack Identity which will query the CS department's LDAP server. Once a user is authenticated, they will be able to create and destroy VMs under their own personal network. Administrators will also be able to manage users and their networks, as well as other people's VMs.

Use Case 3: Personal VMs for Teaching 'Introduction to Linux' Class.



A professor wishes to teach a class on how to use Linux operating system. The professor would like to create a number of VM's for students to work with. This will allow an easy and fast method to create new VM's for any student. The professor would like to have the ability to create and destroy these VM's with relative ease. Each VM also needs to be configured automatically to authenticate against the CS department's LDAP.

Use Case 4: Transition from VMware Infrastructure to OpenStack.

The computer science department already has an existing infrastructure for managing VMs for researchers, professors and students. Some of the users of OpenStack may want to migrate some of their existing VMs to OpenStack. The move should be requested to the system administrator, who is going to run a process to move virtual machines from VMware's VSphere to OpenStack. The owner and permissions should also be migrated to OpenStack.

3. Non-Functional Design

- Must use the CS department’s Active Directory for authentication
- Permissions must be given on a need-to-use basis
- Must use secure communication or the communication must use encryption
- Access to the hardware must be given on a need-to-use basis
- Setting up a new virtual cluster needs to be done through the web interface in the range of minutes
- Must provide documentation for future use and maintenance

4. Definitions and Acronyms

Term	Definition
Backend	A term that refers to a computer that stores data.
CLI	An acronym that stands for Command Line Interface. This is a common term for terminal or command prompt.
Cloud Computing	The use of computing resources that are presented as a service and delivered over a network.
Flavor	An available hardware configuration for a server.
IaaS	Infrastructure as a Service.
Image or System Image	An image is a copy of the software state of a computer system. This includes the operating system and the data.
LDAP	An acronym that stands for Lightweight Directory Access Protocol. LDAP is a protocol that allows user authentication and access restriction to computers, data, etc.
Object Storage	Storage that consists of discrete containers of data with associated metadata, each with a unique identifier.
SAN	An acronym that stands for Storage Area Network, a network of systems that are dedicated to file access and storage.
VM	An acronym that stands for Virtual Machine, software that allows multiple instances of operating systems to run on the same system.

5. Document Review

Project Manager

Name:	John Rensberger
Date:	
Signature:	

Mentor

Name:	Michael Fork, Lance Bragstad, Mathew Odden, Adam Reznechek
Date:	
Signature:	