High Level Design

	Name	Location	Date
Project Title	Cloud Based File Synchronization System	Mumbai	27th December 2012
Team Members	Akshay Jadhav, PranavSalunke	Mumbai	
Guide	Prof. VinitKotak	Mumbai	
Distribution List	Shah & Anchor Kutchhi Engineering College & Mumbai University	**Version No.**	1.0

Table of Contents

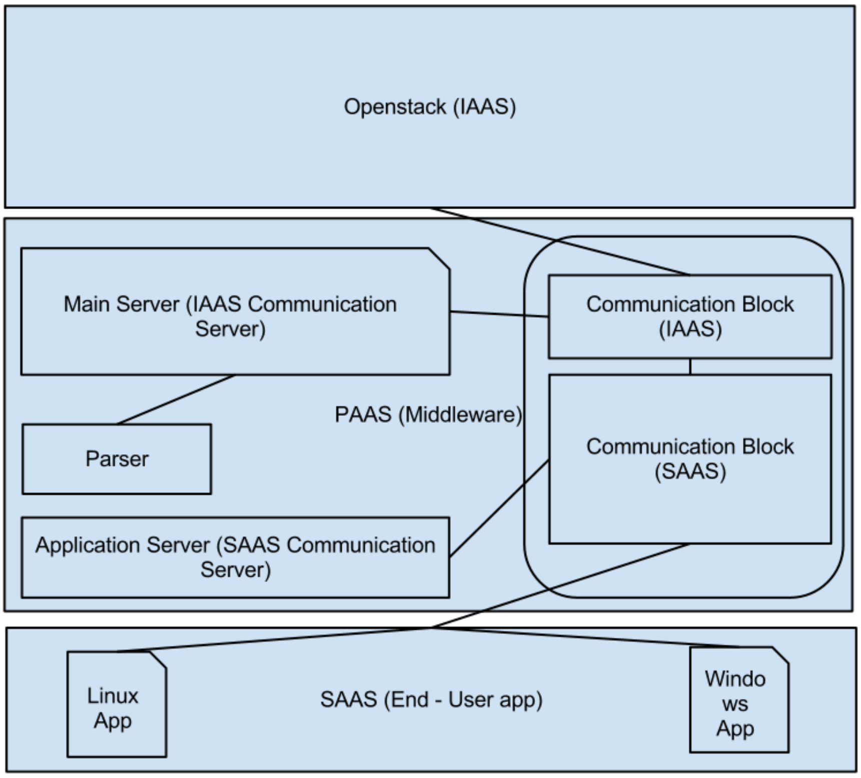
Project Description

A multi-way file synchronization system that enables users to have their files present on their devices. Files are saved on the cloud for access from anywhere at any given time.

The system is designed so as to support a plethora of popular devices widely used by both, business as well as casual users.

Organizations that require a private cloud to be set up for use of this application would be provided the Infrastructure, Platform and Network to seamlessly run the application. Since IaaS, NaaS and PaaS are provided, customization with respect to the infrastructure required to run the application would be possible.

System Architecture


Openstack (IAAS)
Main Server (IAAS Communication Server)
Communication Block (IAAS)
PAAS (Middleware)
Parser
Communication Block (SAAS)
Application Server (SAAS Communication Server)
Linux App
SAAS (End - User app)
Windo ws App

Openstack (IAAS)

A Cloud Operating System that's used to produce the ubiquitous open source cloud computing platform for the system.

For the system developed, it is used to control large pools of compute, storage and networking resources throughout a datacenter, all managed through a dashboard that gives administrators control while empowering them and developers to provision resources through a web interface.

Being easily compatible with Amazon's EC2 Cloud, the Cloud can easily be scaled up and be converted to a Hybrid Cloud.

Middleware (PAAS)

The tier is developed so as to handle computation and communication required to synchronize users' files. Major functioning of the system is handled by this tier and the end-user application is light and thus, extremely portable.

Platform independence is taken care of; thus making the system compatible with various mobile platforms and making it mobile in true sense.

End-User App (SAAS)

The end-user application developed for systems running Ubuntu 10.10+ provides a sophisticated approach to users to manage files that are to be pushed to the Cloud and synchronized over other devices.

Environment Specification

Server Side: Middleware

Operating System: Linux Ubuntu 12.04/12.10 Server

Processor: Intel Chipset/AMD Chipset (VT Enabled) Memory: 8GB

Hard Drive Capacity: 500 GB expandable

Server Side (IaaS)

Cloud Controller Node (Runs network, volume, API, scheduler and image services)

Operating System: Linux Ubuntu 12.04/12.10 server Processor: Intel Chipset/AMD Chipset (VT Enabled) 64-bit x86 - Two NICS

Memory: 12GB

Hard Drive Capacity: 30GB (SATA or SSD)

Volume Storage: Two disks with 2TB (SATA) for volumes attached to the compute nodes.

Network: one 1GB Network Interface Card.

Memory: 16GB

Compute Node (Runs virtual instances)

Operating System: Linux Ubuntu 12.04/12.10 server Processor: Intel Chipset/AMD Chipset (VT Enabled) 64-bit x86 – Two NICs

Memory: 32GB

Hard Drive Capacity: 30GB (SATA or SSD) Volume Storage: Two disks with 2TB (SATA) for volumes attached to the compute nodes.

Network: Two 1GB Network Interface Cards.

Memory: 16GB Hard Drive Capacity: 750GB expandable

Client Side

Operating System: Linux

Processor: Intel Chipset/ AMD Chipset

Memory: 256 MB

Hard Drive Capacity: 60GB

Development Tools:

Eclipse Indigo (Mac OS X 10.7.5)

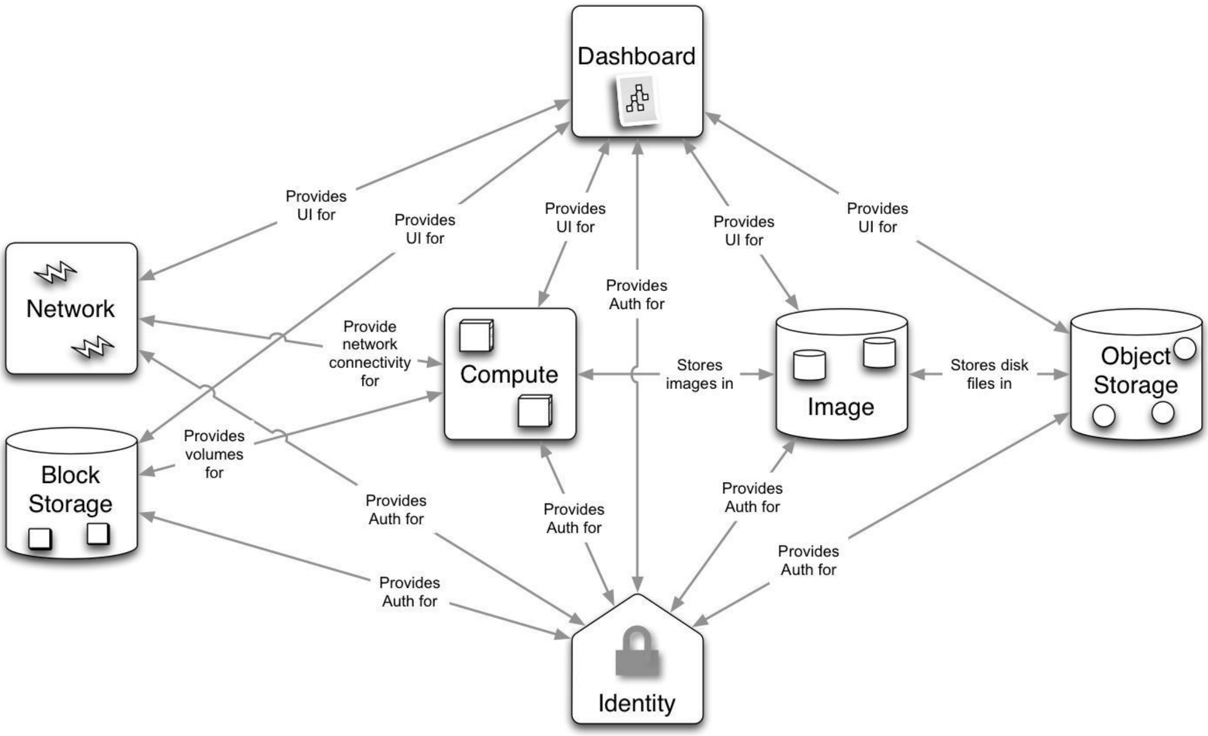
Netbeans IDE 7.2 (Ubuntu 12.10)

Documentation Tools:

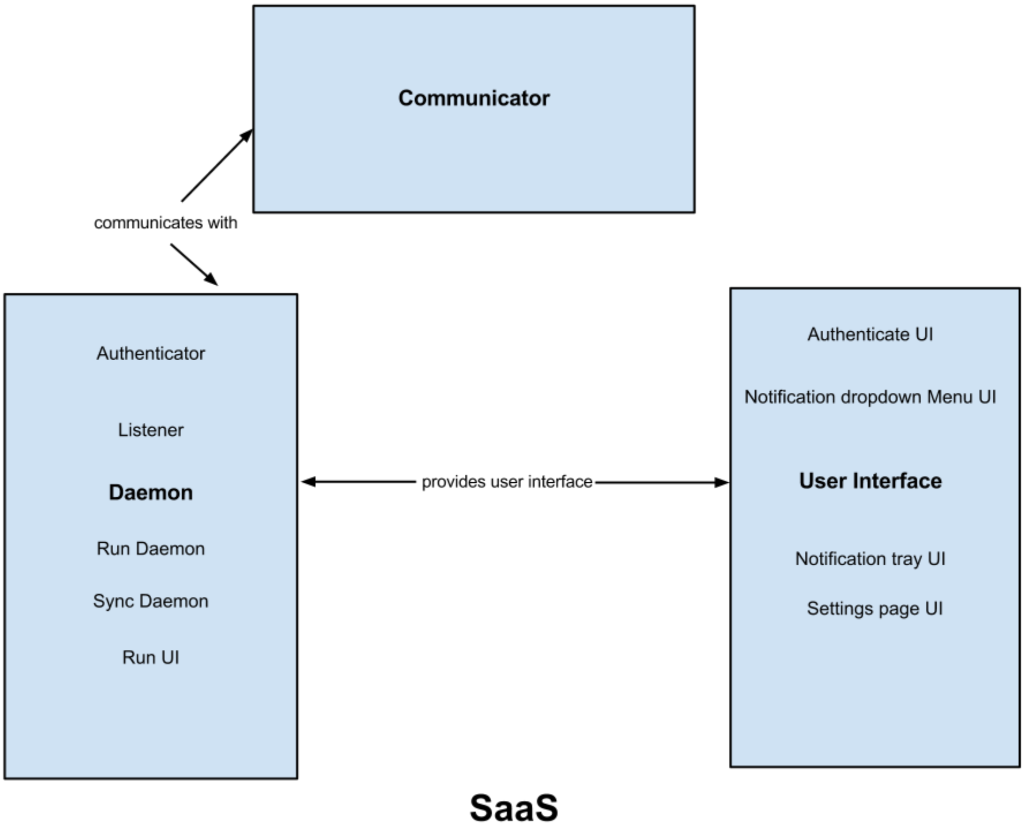
OpenOffice.org 3
OpenProj

Module Breakdown

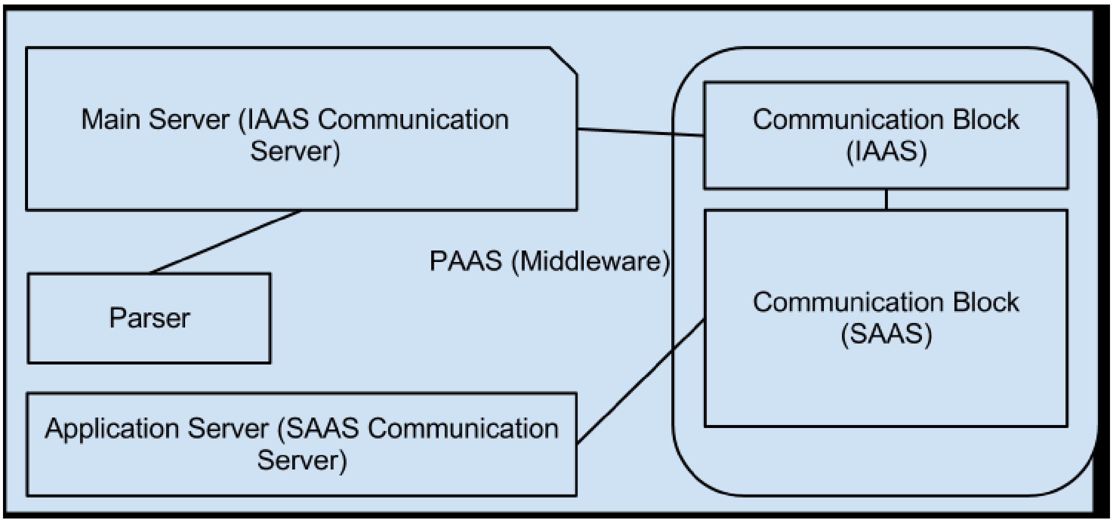
OpenStack (IAAS)



User Application (SaaS)



Middleware (PaaS)


Main Server (IAAS Communication Server)
Communication Block (IAAS)
PAAS (Middleware)
Parser
Communication Block (SAAS)
Application Server (SAAS Communication Server)

Design Description

OpenStack (IaaS)

- Object Store (codenamed "Swift") provides object storage. It allows you to store or retrieve files (but not mount directories like a fileserver). Several companies provide commercial storage services based on Swift. These include KT, Rackspace (from which Swift originated) and Internap. Swift is also used internally at many large companies to store their data.
- Image (codenamed "Glance") provides a catalog and repository for virtual disk images. These disk images are mostly commonly used in OpenStack Compute.
- Compute (codenamed "Nova") provides virtual servers upon demand.
- Dashboard (codenamed "Horizon") provides a modular web-based user interface for all the OpenStack services. With this web GUI, you can perform most operations on your cloud like launching an instance, assigning IP addresses and setting access controls.
- Identity (codenamed "Keystone") provides authentication and authorization for all the OpenStack services. It also provides a service catalog of services within a particular OpenStack cloud.
- Network (codenamed "Quantum”) provides "network connectivity as a service" between interface devices managed by other OpenStack services.

User Appication (SaaS)

Communicator

The communicator is basically a simple client socket program which will synchronize different blocks of files and folders as per the PAAS server's call.

Daemon

Daemon consists of a set of background processes that monitor/perform tasks related to file activity and are responsible for the automatic synchronization of a user's files from both – client machine to cloud storage and cloud storage to client machine.

The Authenticator Daemon performs the task of authenticating the user based on his credentials.

The Listener Daemon performs the task of monitoring the changes in files on the cloud and the user's machine(s).

The Run Daemon performs the task of kickstarting the SaaS modules.

The Run UI Daemon performs the task of invoking various UI programs.

The Sync Daemon performs the task of synchronization of files between the client machine and the cloud.

User Interface

The User Interface is responsible for providing the user interface that a user would need for getting started as well as monitor file activity.

Authenticate UI provides the user interface for the initial authentication that the user needs to go through to get started.

Notification dropdown Menu UI provides the user interface for the dropdown menu that contains various options related to modifying the app functioning as per user's requirements.

Notification Tray UI handles the user interface that exists to notify the user of file activity in almost real time.

Settings page UI handles the settings that a user needs to perform to handle files as per his requirements.

Middleware (PaaS)

The Parser is the component responsible for processing the XML files and forming a folder structure on both the Cloud storage system as we as on the User's machine.

The IaaS Communication block is responsible for handling communication between the IaaS Module and the Communication block of the SaaS Module. It listens for any changes in file/folder structure and sends a request to retrieve/push the latest copy of the file changed.

The SaaS Communication Block hands communication between the Application server and the IaaS Communication block.

The Application Server tracks changes in the user’s file system and sends triggers to the IaaS module to update the files. It also sends a request to re-parse the XML file so as to have the latest version.