CMPE 283
Virtualization Technologies

Project 1: Availability Manager

Moksha Bhargav Vanam
SJSU ID: 010011638
mokshabhargav.vanam@sjsu.edu

Spring 2015

Submitted To

Professor Simon S.Y. Shim

Computer Engineering Department

San Jose State University

Table of Contents

1. Introduction

1.1 Goals

- The main goal of this project Disaster Recovery System is to acquire experience on working with several hypervisors and management servers.
- To explore the capabilities of the components of hypervisor, ESXi Java APIs and learn about the issues arising from interoperability.
- To apply these concepts to a real-world problem and learn and debug the interoperability issues in real-time.

1.2 Objectives

- The main objective is to develop an availability manager that acts as a virtual monitor which checks the status of the virtual machines running on the hosts in a VCenter.
- Virtual Machine state should be captured in snapshots at regular 10-minute intervals if the virtual machine responds to the availability manager.
- If the Virtual machine is not reachable, then availability manager detects the failure and recovers it.

1.3 Needs

- The Availability Manager checks the life of the virtual machines so we need virtual machines.
- To have virtual machines we need to have virtual hosts.
- VCenter is needed to control the virtual hosts and the virtual machines.

1.4 Background

- VMware provides a centralized application VCenter server which manages the servers and the virtual machines associated to them.
- VMware VCenter Server provides a unified view of network by managing multiple ESX servers and virtual machines from different ESX servers through a single console application.
- VCenter Server is responsible for several operations such as creating data centers, adding hosts to data center, creating resource pools within hosts and creating VMs on the host and allocating to logical resource pool.
- Although VCenter server provides management and monitoring interface, we need a separate module which monitors the states of host and VMs continuously and replaces them with working VMs in case of failure.
- Therefore to deal with the Virtual machine failures we build a availability manager that monitors the status, detects the failures and restart them on another active host.

2. Requirements

2.1 Functional requirements

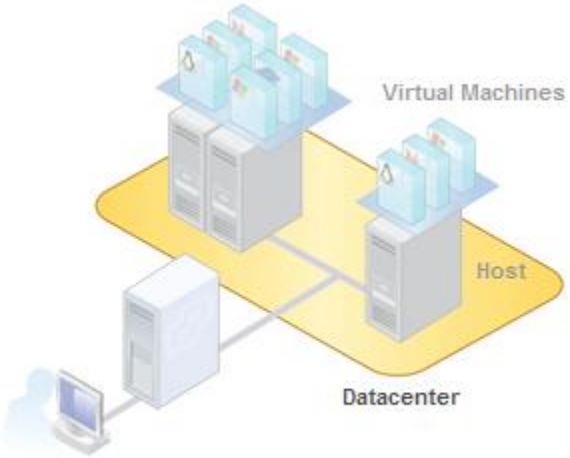
- The virtual machine status is said to be live if it responds to the pinging.
- If the pinging fails then the virtual machine should be checked for its power status i.e. its powered on or off.
- The Snapshot should be done at regular intervals and only one snapshot should be maintained at any point of time.
- If user powers of virtual machine, an alar should be generated.
- The application should gather all statistics such as CPU, I/O, and Network etc. for a Virtual Machine and display it in a text format.
- The application should be able to detect whether the virtual machine is powered off or powered on and not reachable.
- The application should be able to recover virtual machine at any circumstances.

2.2 Non-functional Requirements:

- System should be reliable.
- System should ping the VM at particular time interval.
- It should not migrate the VM that is powered off or shut down by the user.
- There should be some time gap between two activities of a VM, because after the VM is started it takes a while to start the network and make the VM totally functional.
- Performance for monitoring should be good.

3. Design

3.1 Architecture



3.2 Major Components

- VMWare ESXi managed by VMWare VCenter Server
- Host running on a centralized datacenter.
- Virtual machines on hosts.
- Operating system like Ubuntu32 installed in virtual machines.
- VMWare tools, which need to be installed on operating system of the virtual machines in order to get network information of virtual machine.

3.3 Key workflows

- The availability manager (Main.java) has two threads running continuously, one for pinging the virtual machines and performing the recovery management, the other one for taking the snapshot of the virtual machine.
- Now there can be two scenarios if the system is not able to receive ping-replies from VM. Either the user powers off the VM explicitly, or the VM is in running state and still there is no connectivity. If the user powers it off, then it is not considered to be a failure and it waits for the user to start the VM. If there is no connectivity, then it checks with another host for its health by pinging it, and with the help of previous snapshots, it will clone that VM and cold migrate it to that host using VMDK format.

4. Implementation

Environment: Eclipse IDE
Tools: VSphere client
Scripting languages and tools: JAVA

4.1 Screenshots:

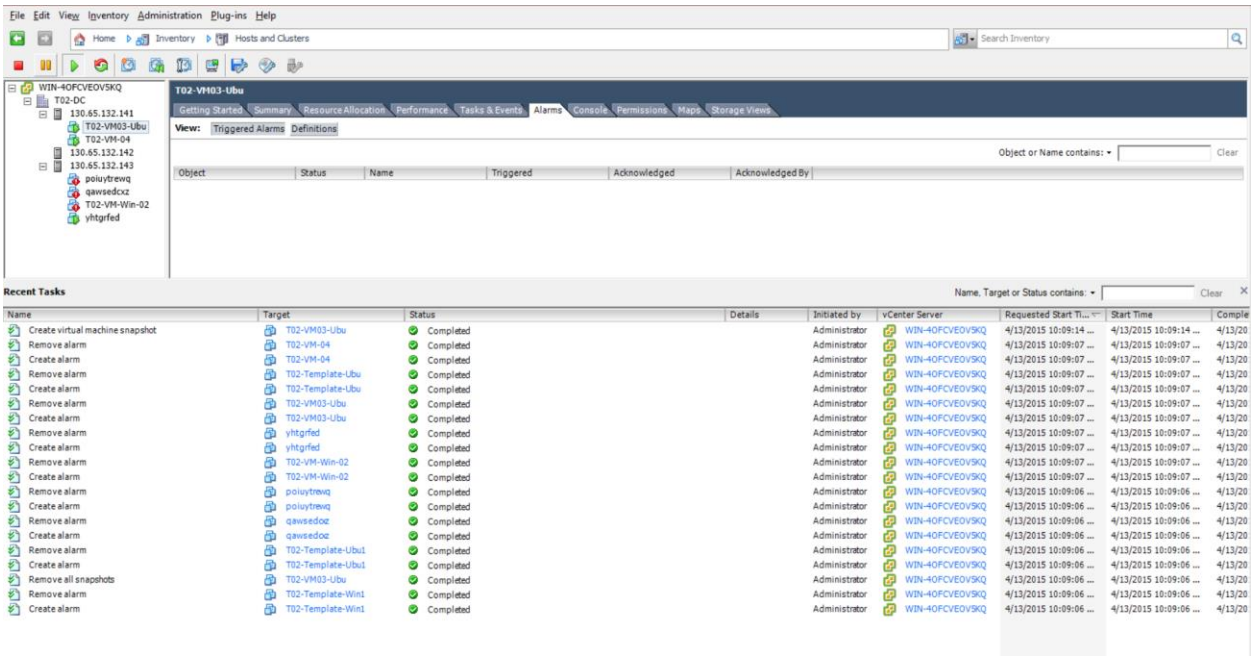


Figure 1: An alarm is created when the Availability Manager (main.java) starts running

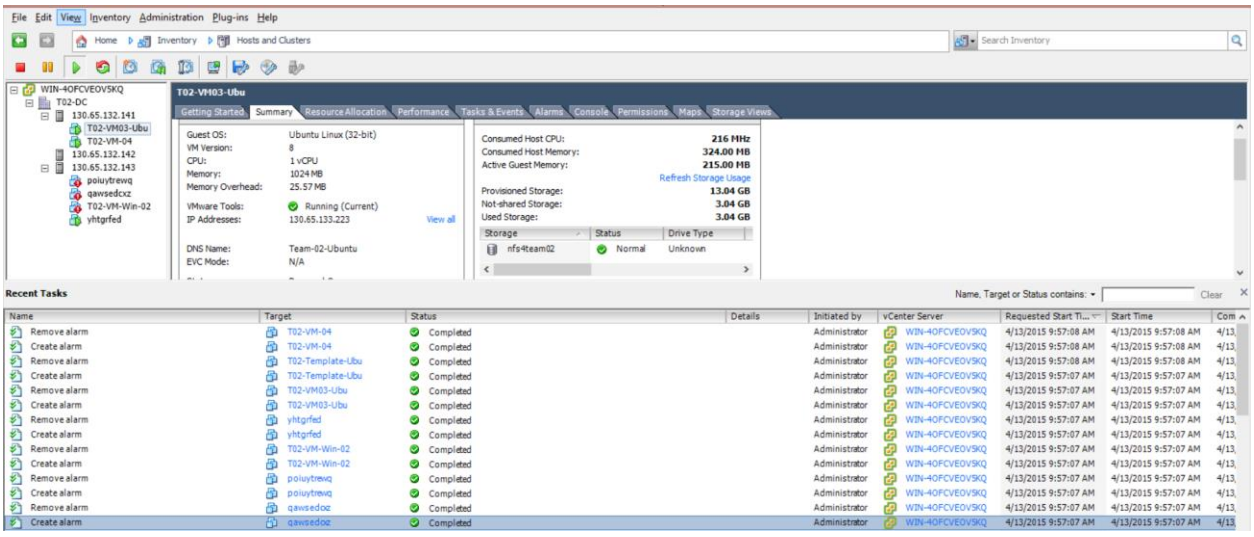


Figure 2: The statistics of the virtual machine are displayed

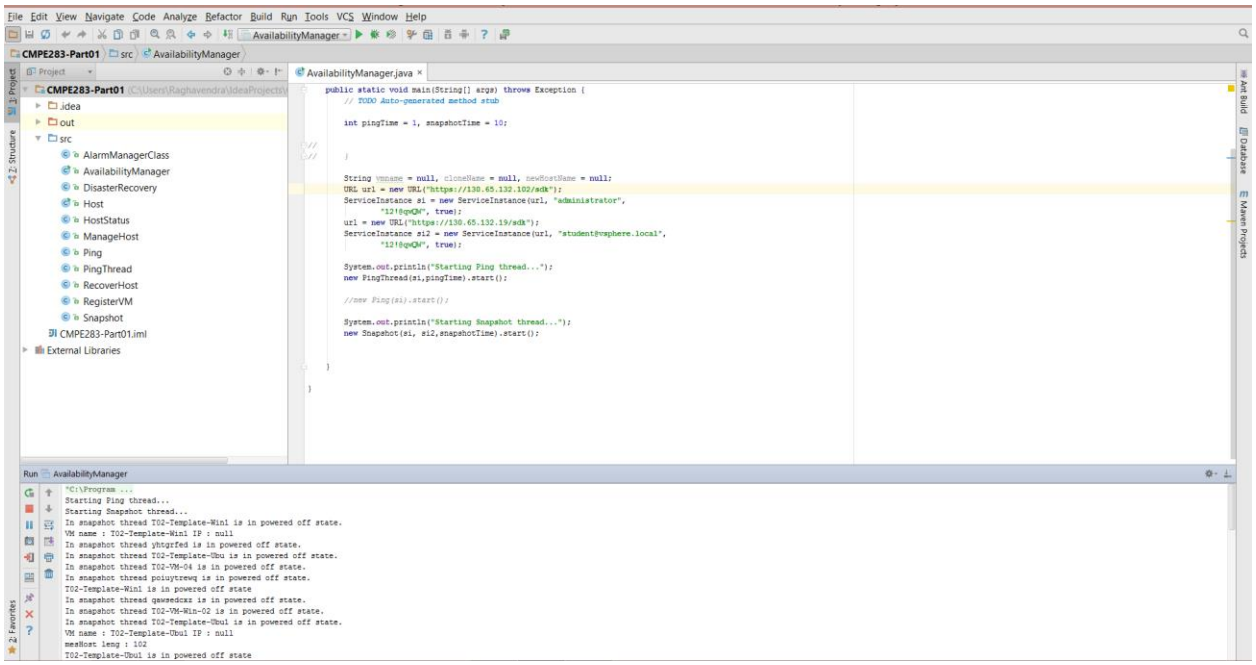


Fig VM failed but host is reachable, Availability manager recovered VM from latest snapshot.

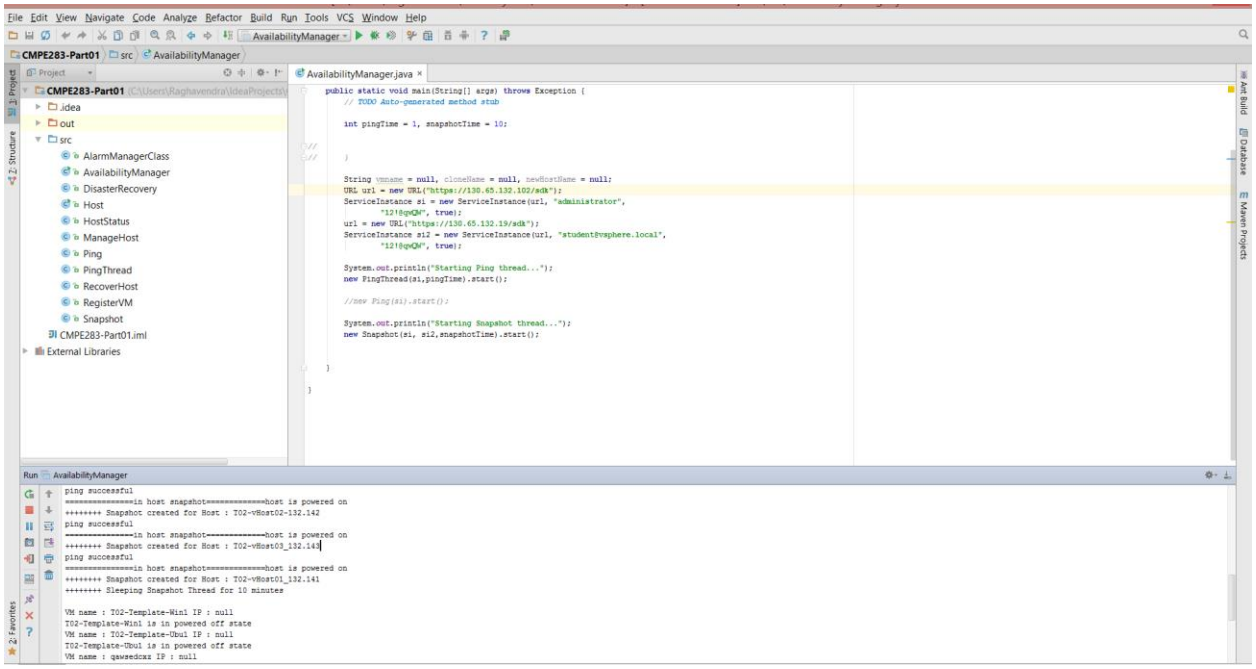


Fig 6: VM and host both failed, Availability manager recovered VM by restoring host from its latest snapshot and starting virtual machines running on that host.

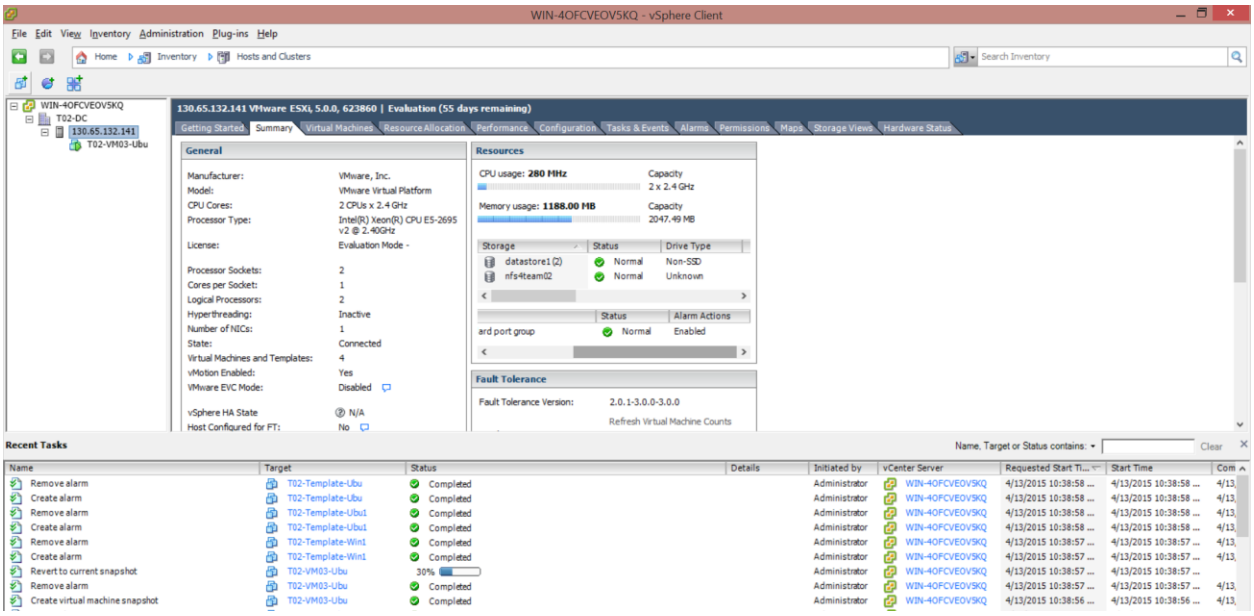


Fig : Image showing process of recovering host from its latest snapshot available. Finally all virtual machines are available and running.

5. Class Diagram:

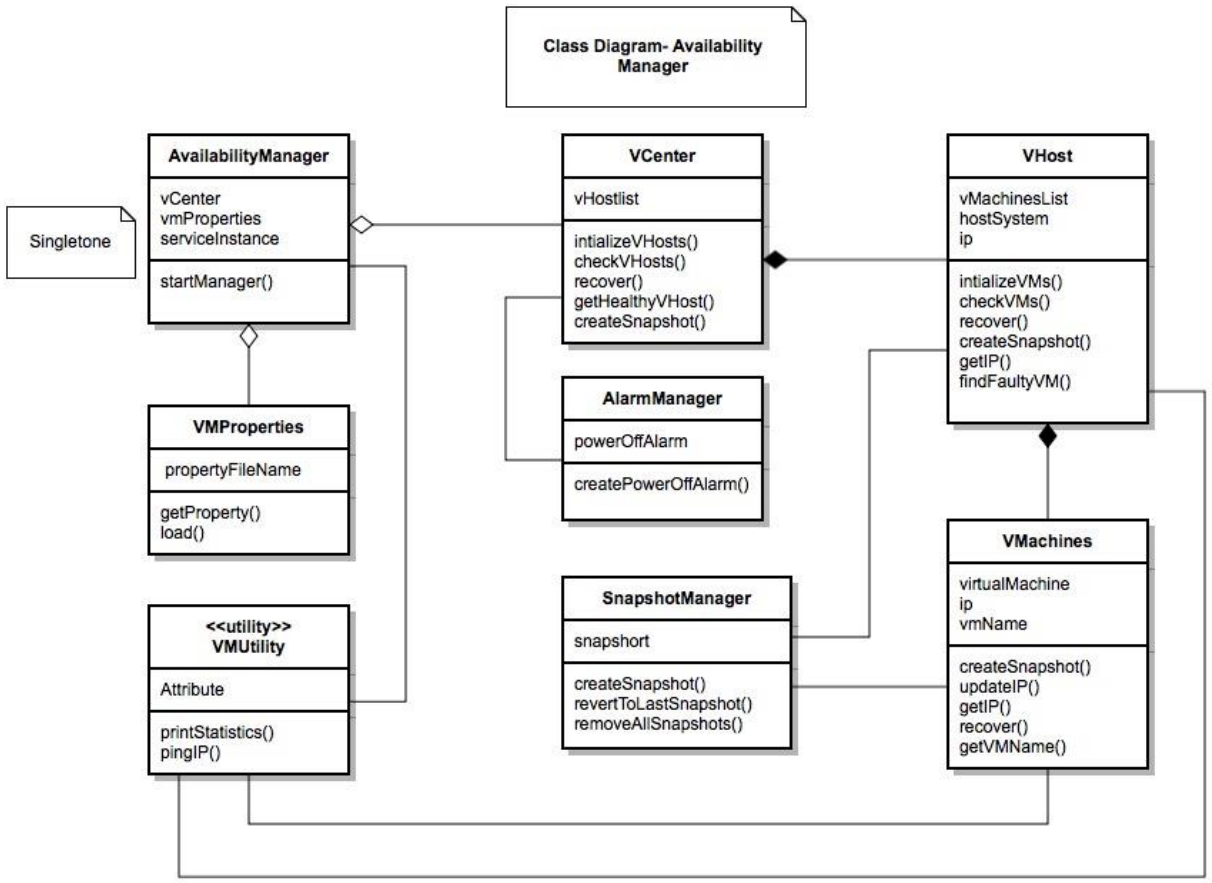


Figure 3: Class Diagram

6. Answers to questions

1. Briefly explain the design of your Availability Manager with the help of a class diagram. Also explain the number of threads you've used for the Availability Manager.

Ans:

- Availability Manager manages sessions and threads to continuously check the status of VMs and VHost for pinging and also for snapshot.
- Alarm manager alarms when user powersoff the Virtual Machine
- Disaster Recovery recovers the latest snapshot for the Vhost and VM which is unable to retrieve ip address of VMs and Vhost
- VCenter manages all virtual hosts in the system and does the operations like checking health of vHosts, creating snapshots for vHosts.
- Snapshot Manager does all tasks related to snapshots like creating snapshots, removing snapshots and reverting virtual machine to latest snapshot.

- Once the program started, Availability manager starts loads all virtual hosts and virtual machines and starts three threads.
- One thread takes snapshots of virtual machines and virtual hosts, it monitors the state of virtual machines takes necessary actions if any virtual machine is not reachable.
- All thread run continuously and time interval for them to sleep is configurable though a property file.
- In case of virtual machine failure, availability manager checks the status of the virtual host. If host is available, availability manager recovers the virtual machine from the latest snapshot available.
- In case host is also failed, host will be recovered from the latest snapshot available.

2. How does your availability manager handle the scenario where in the vHost itself is found not to be alive?

Ans:

Availability manager checks for health of virtual machines, if a virtual machine is found to be not reachable, availability manager checks for the status of vhost. If host is not reachable, availability manager recovers the virtual host from the latest snapshot and starts all virtual machines running on the host.

3. In case of failure, what is a good approach during Disaster Management of Virtual Machines?
    - Check the Host first, then the Virtual Machine.
    - Check the Virtual Machine, then the Host.

   Justify your answer with sufficient reasons.

Ans:

Checking the Virtual machine first and then the host is a good approach in Disaster Management of Virtual Machines. The major idea of Disaster Management is to monitor the Virtual machines and not the hosts. The other way may result is redundant and unnecessary actions. For example consider if host and virtual machines both are running, if we check virtual machines first and then hosts, checking virtual machines health is completed with first level it self. We won't check health of hosts. Incase if we check host first, then we have to check virtual machines even though host is running fine causing one more unnecessary actions.

7. Discussion

- **The host add/remove mechanism**

  Host add/remove is a part of the application Availability Manager. This mechanism is achieved using the below command

```
ManagedEntity [ ] hostList = new InventoryNavigator
(rootFolder).searchManagedEntities (new String [ ][ ] { {"HostSystem", "name" }, },
true);
ManagedEntity[ ] virtualMachines = new InventoryNavigator(rf).searchManagedEntities(
new String[][] { {"VirtualMachine", "name" }, }, true);

Task task = vm.migrateVM_Task(cr.getResourcePool(),newHost,
VirtualMachineMovePriority.highPriority, VirtualMachinePowerState.poweredOff);
```

Manually we can add and remove hosts in the VCenter . By deploying the student template we can create a host and we need to change the network settings of the host. And the addition and removal is done through VSphere client by right clicking on the data center associated with the VCenter.

- **The approach used to configure the failure detection for each VM**

  The approach followed by the Availability Manager for failure detection of Virtual machine is to first check the VM, if the VM is inactive, then we check for the power status of the VM and if its powered off an alarm will be generated else checks for the host status. If host is alive, virtual machine will be recovered from the latest snapshot of the virtual machine. If host is not reachable, host will be recovered from the latest snapshot of host and virtual machines will be turned on.

- **How host failures were detected**

  The host failures are detected by pinging the host ip address. This detection is done after checking the Virtual Machine status.

- **The mechanism used to convert between the image formats used by the hypervisors**

  The hypervisor uses a software called VMConvertor which is used to covert between different image formats. The VMConvertor follows a simple mechanism which includes 3 steps.
  We need to specify virtual machine or third party to convert. Next we need to specify the destination format, virtual machine name, and location for the new virtual machine to be created. Create/Convert to destination virtual machine and configure it be created.

8. Conclusion

The application Availability Manager successfully implements a Disaster Recovery system, which continuously monitors the status of the virtual machines running on any host, and recovers virtual machine. In addition to this, it also prevents a failover of a virtual machine if a user powers it off explicitly.

References

- http://vijava.sourceforge.net/doc/getstarted/tutorial.htm