IAS - Group - 3 (Team - 3)

Requirement Document

Deployment Manager, Node Manager, Fault Tolerance and Monitoring

Team members:

Yash Pathak (2022201026)

Vivek Kirpan (2022201071)

Prasad Kawade (2022202006)

Contents

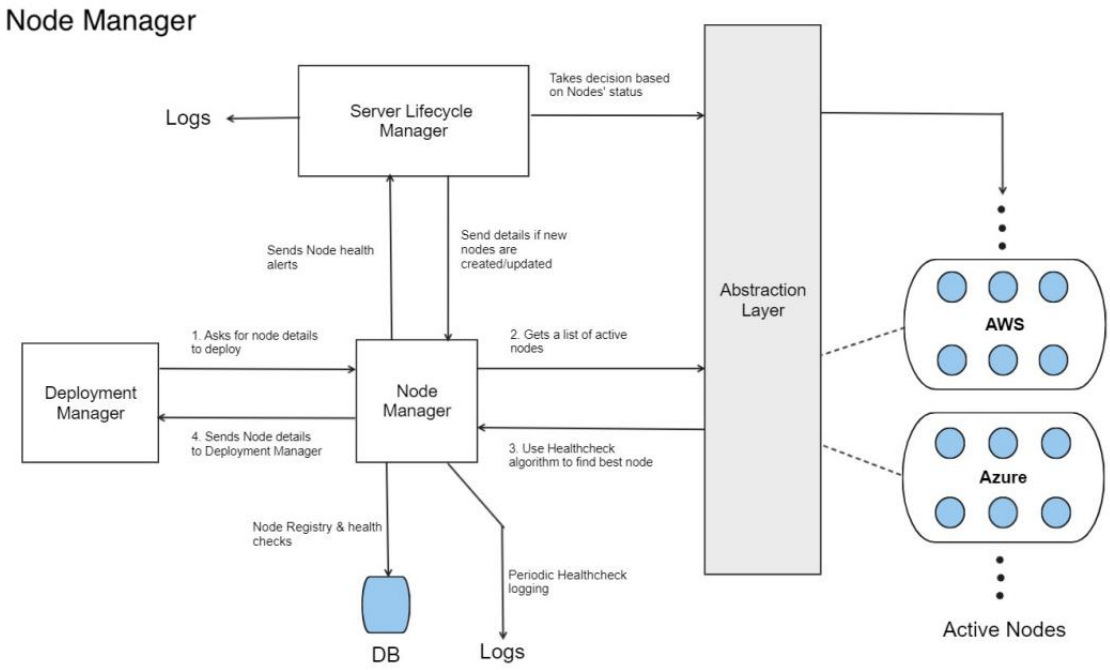
Node Manager

1.1 Description

Node manager module that is responsible for managing and allocating new nodes, which are virtual machines, based on requests. The module provides node details, including a unique node ID, node IP when a node request is made. If the available nodes are not sufficient to satisfy the request, the module requests the server lifecycle.

It uses an algorithm based on VM's disk, memory and CPU usage to provide the healthiest VM for deployment when requested. It also logs the health of VMs constantly.

1.2 Flowchart



1.3 Subsystems

- Nodes - Nodes can be either physical servers or virtual machines, and they are responsible for executing platform components or application instances. Additionally, nodes can also communicate with the sensor manager to send or receive data from sensors.

- Node Manager - The Node Manager module is tasked with managing node-related information and providing a new node to the deployer if necessary.

- Shared DB & Logs – The shared DB holds the VM pool which is managed by Server Lifecycle manager. The logs in the shared filesystem holds the VM health logs for monitoring purposes.

1.4 Functional Requirements

- The Node Manager module is responsible for supplying the deployer with a new node whenever it is needed.
- Applying health check algorithm to find out the healthiest node.
- Logging the health of VMs.

1.5 Interaction with other modules

- Sensor Manager - If an application instance is running on a specific node, it may need to communicate with the sensor manager to send or receive a data stream from one or more sensors.
- Deployment Manager - If the load on currently active nodes increases, the deployer may request a new node from the Node Manager to handle the additional load.
- Monitoring and Fault tolerance - The system sends a health message to the Monitoring and Fault Tolerance module.
- Server Lifecycle Manager – The system interacts with Server Lifecycle manager when there are no nodes in the VM pool. Server Lifecycle manager handles the VM pool and creation/updating of nodes.

1.6 Test Cases

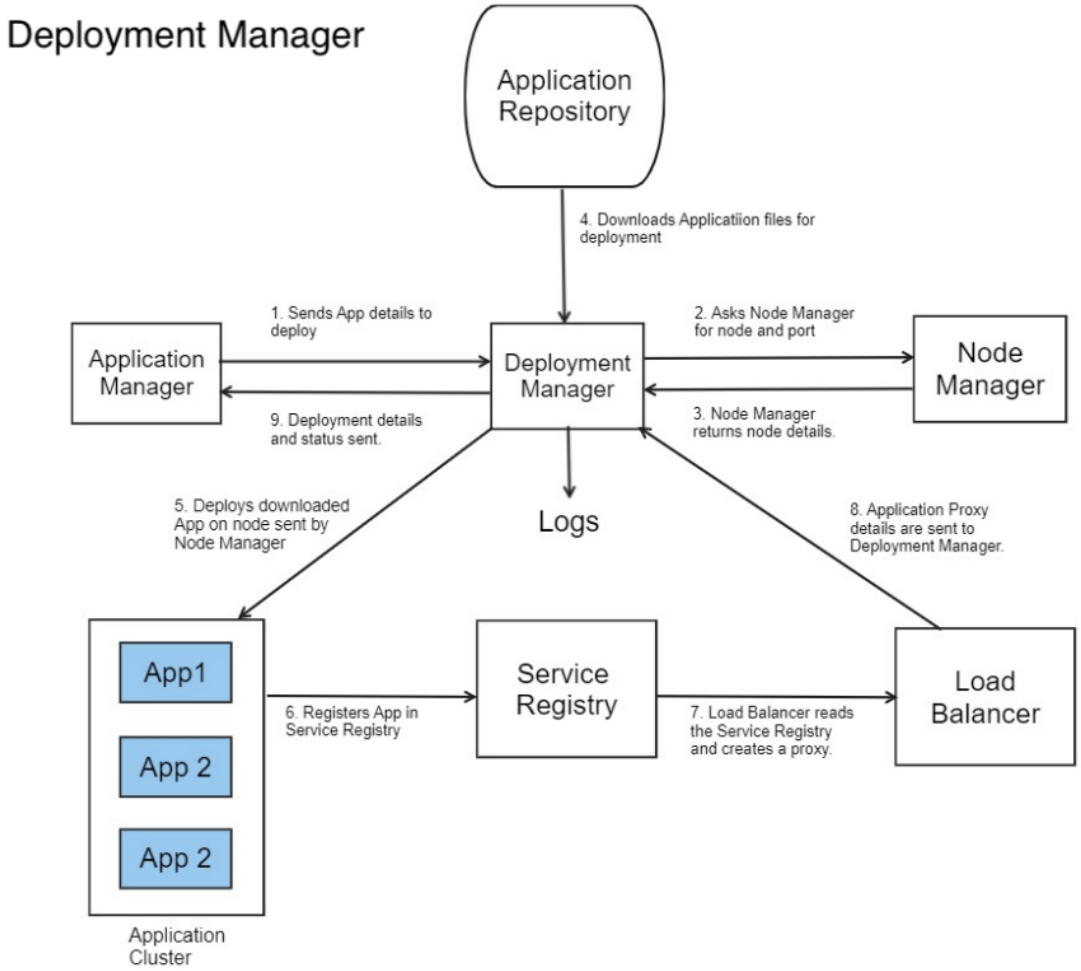
Input: If the available nodes are occupied and a deployment manager is asking for the execution of an algorithm.
Output: New node id is first asked from the server lifecycle manager. It added it to the pool. Node manager runs the health check algorithm and returns the best VM.

Deployment Manager

2.1 Description

Deployment manager module is responsible for deployment of new application instances on the VMs allocated by the Node manager. It requests the Node Manager for new node on which it can deploy the new application instances. It connects to the shared filesystem and to the application repository to get the application artifacts. It generates the files for deployment and runs iton the allocated VM for the deployment of the application.
It then registers it in Service registry and sends requests to the Load balances to create a proxy for the deployed application. This is then returned to the Application Manager.

2.2 Flowchart



2.3 Subsystems

- Shared filesystem connector – This subsystem helps to connect to the shared filesystem to get the application artifacts and push to the allocated VM for deployment.

- Deployment Manager - The Deployment Manager module is tasked with deployment of the application instance requested by the Application Manager.

- Deployment Files Generator – This subsystem helps in generating the files dynamically for each application that needs to be run for deploying that application.

2.4 Functional Requirements

- The deployment manager deploys the application instance requested by the Application manager.
- The deployment manager fetches the application artifacts from the shared filesystem.
- The deployment manager dynamically generates the deployment files for each application.
- The deployment manager has a heartbeat API which responds to Fault Tolerance manager.

2.5 Interaction with other modules

- Application Manager – Application manager requests deployment of an application instance by sending the application details Deployment Manager after deploying and getting proxy from Load balancer, returns the IP and port for the deployed application instance.

- Node Manager – When an application deployment request comes, it requests the Node manager to provide with a node. Node manager provide unique VM with IP.

- Load Balancer – Once the application is deployed, it requests the Load Balancer to create a proxy for this deployed application.

- Service Registry – It does service registry for the deployed application.

- Fault Tolerance and Monitoring – It should response to the heart beat API request by the Fault Tolerance and monitoring.

2.6 Test Cases

Input: Application manager send a deployment request to deployment manager.
Output: New node id is first asked from the Node manager. Deployment manager then does its tasks and deploys the application.

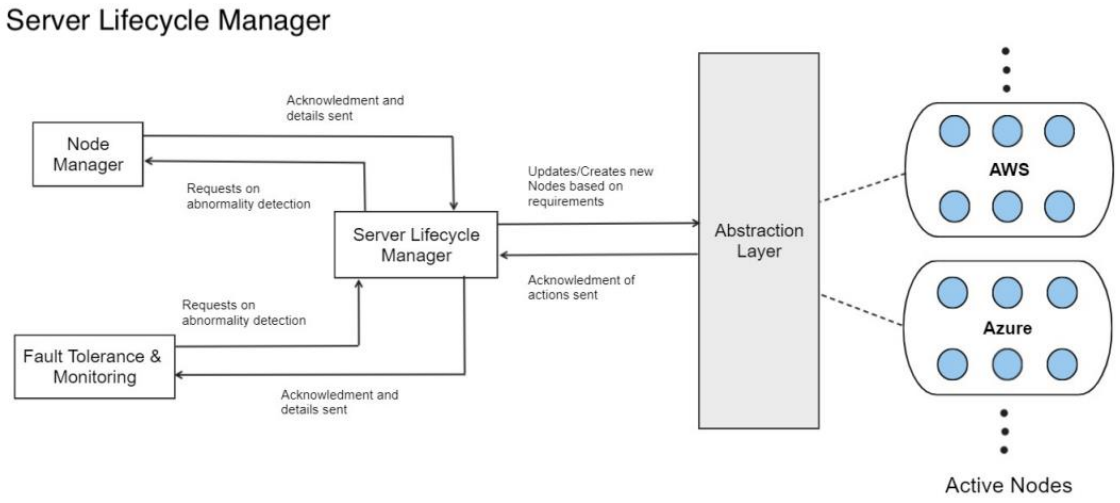
Server Lifecycle Manager

3.1 Description

Server Lifecycle Manager module is responsible for managing the nodes (VMs) and server pools which is maintained in the shared DB. It can add new servers to the pool, update existing servers, create new servers and manage the pool.

Whenever there are no new/healthy servers available for Node manager based on its health check, Server lifecycle manager should update the VMs if necessary or add new VMs in the pool.

3.2 Flowchart



3.3 Subsystems

- Shared DB connector – This subsystem helps to connect to the shared DB to get the VM pool and VM details for managing and updating it based on the operations done the Server Lifecycle manager.

- Server Lifecycle Manager - The Server Lifecycle Manager module is tasked with managing the server pool and the servers. It also helps in addition to the server pool, create new servers or update the server based on the requirements.

- Service Provide Interface – This subsystem helps in connecting to the service provider console to update and create new VMs on the go.

3.4 Functional Requirements

- The Server lifecycle manager should be able to update the server pool whenever required by the Node manager.
- The Server lifecycle manager should be able to update/ patch update the VMs whenever required by connecting to the service provider console.
- The Server lifecycle manager should be able to create new servers whenever required by connecting to the service provider console.

3.5 Interaction with other modules

- Node Manager – When an application deployment request comes, it requests the Node manager to provide with a node. Node manager provide unique VM with IP.

- Service Registry – It does service registry for the deployed application.

- Fault Tolerance and Monitoring – It should response to the heart beat API request by the Fault Tolerance and monitoring.

3.6 Test Cases

Input: Node manager is not able to get a healthy node and cannot send it to the deployment manager for deployment.
Output: New nodes are added to the pool by the Server Lifecyle Manager and existing nodes are checked for updates.