

## CS 446 Homework 3

### Chapter 5

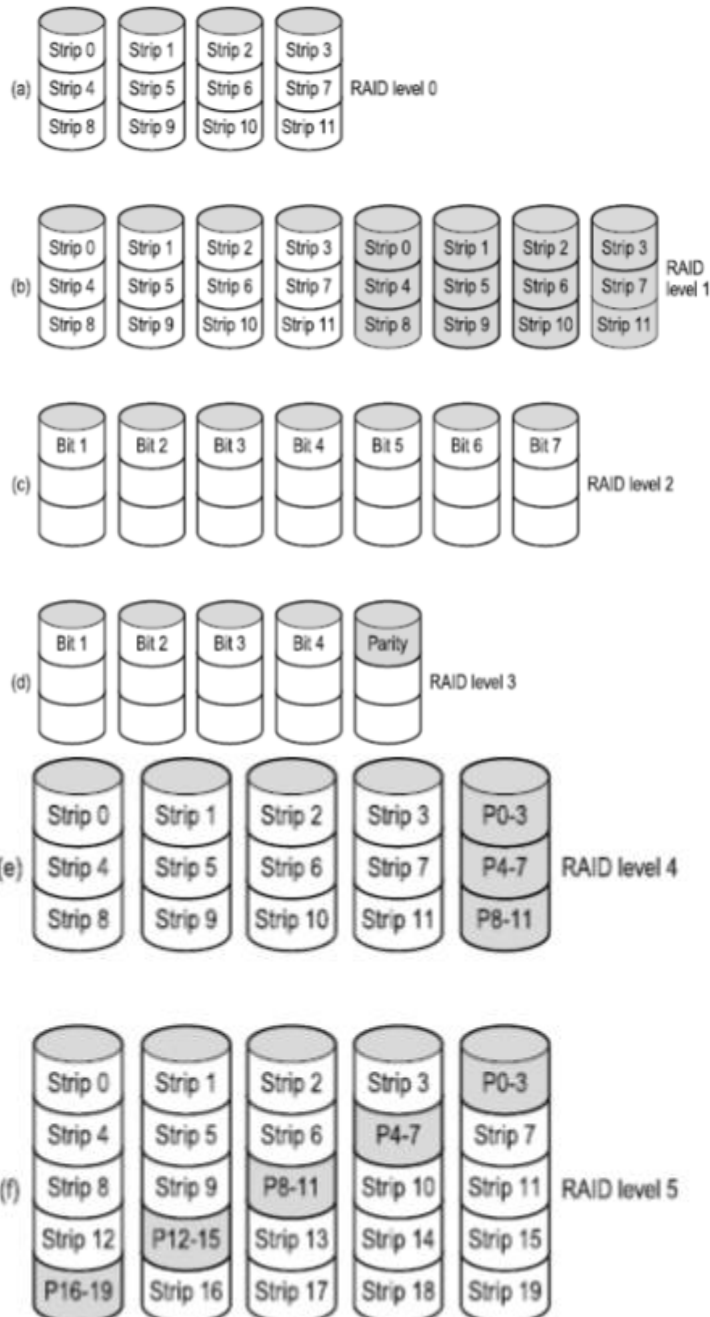
1. Compare RAID level 0 through 5 with respect to read performance, write performance, space overhead, and reliability.

	Read Performance	Write Performance	Space Overhead	Reliability
<b>RAID 0</b>	<i>High Performance; parallel and simultaneous read operations</i>	<i>High Performance; randomized and concurrent writes</i>	<i>Requires 2 drives minimum. Data is striped and written onto multiple drives. No space overhead</i>	<i>Offers no support for data redundancy, fault tolerance, or error detection. Failure will result in full data loss.</i>
<b>RAID 1</b>	<i>Fast read speed as either disk can be used to read data; read operation double of RAID 0</i>	<i>Moderate performance as both drives need to be written with same data; bottlenecked by slower drive</i>	<i>Requires 2 drives minimum. Data is mirrored on both drives. 100% space overhead</i>	<i>Offers data redundancy and fault tolerance; if any one drive fails the other will be able to take over.</i>
<b>RAID 2</b>	<i>High Performance; parallel read operations</i>	<i>Moderate performance: similar speeds to other RAID groups</i>	<i>Minimum 3 disks. Bit level stripping with parity. Parity information stored on 3<sup>rd</sup> disk. 6 parity drives → 18.75% space overhead</i>	<i>Offers error detection with Hamming code. Detect and correct a single random bit error.</i>
<b>RAID 3</b>	<i>High Performance; parallel read operations</i>	<i>Moderate performance: similar speeds to other RAID groups</i>	<i>Minimum 3 disks. Uses byte level stripping with parity. Parity information stored on 3<sup>rd</sup> disk. 32-bit data word → 3.13% space overhead</i>	<i>No redundancy if parity drive crash; Detect single bit error</i>
<b>RAID 4</b>	<i>High Performance; parallel read operations</i>	<i>Moderate Performance: parity information needs to be stored</i>	<i>Minimum 3 drives. Uses block level stripping. 33 drives assumption → 3.13% space overhead</i>	<i>Provides redundancy; Detect single bit error</i>

## RAID 5

<p><b>High Performance:</b> better read speed than RAID 2-4 (sequential read)</p>	<p><b>High Performance:</b> better write speed than RAID 2-4; but slowed due to parity creation (sequential writes)</p>	<p>Minimum 3 drives. 33 drives assumption → 3.13% space overhead</p>	<p>Parity data are spread across other drives allowing recovery of data; detect single bit error ; high reliability</p>
---	---	--	---

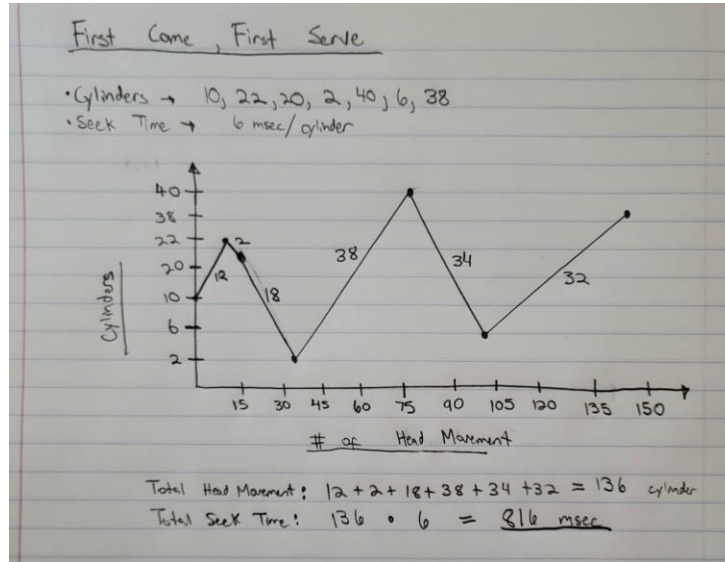
**\*Write performance should be similar or close for ALL RAID levels.**



2. Disk requests come into the disk driver for cylinders 10, 22, 20, 2, 40, 6, and 38, in that order. A seek takes 6 msec per cylinder. How much seek time is needed for

(a) **First-come, first-server**

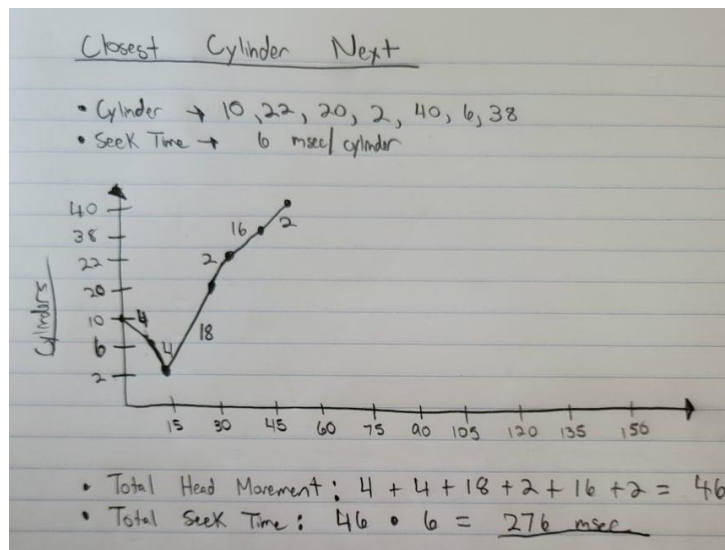
- a  $|10 - 22| + |22 - 20| + |20 - 2| + |2 - 40| + |40 - 6| + |6 - 38| = 136$   
b **Total Seek Time = Head Movement \* Seek Time =  $136 * 6 = 816$  msec**



c

(b) **Closest cylinder next**

- a  $|10 - 6| + |6 - 2| + |2 - 20| + |20 - 22| + |22 - 38| + |38 - 40| = 46$   
b **Total Seek Time = Head Movement \* Seek Time =  $46 * 6 = 276$  msec**

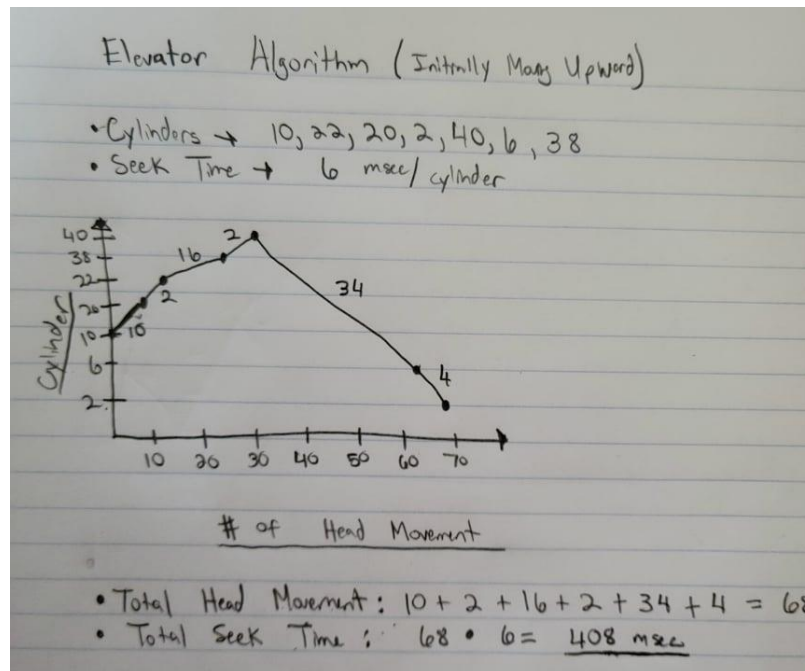


c

(c) **Elevator algorithm (initially moving upward)**

- a  $|10 - 20| + |20 - 22| + |22 - 38| + |38 - 40| + |40 - 6| + |6 - 2| = 68$

b  $\text{Total Seek Time} = \text{Head Movement} * \text{Seek Time} = 68 * 6 = 408 \text{ msec}$

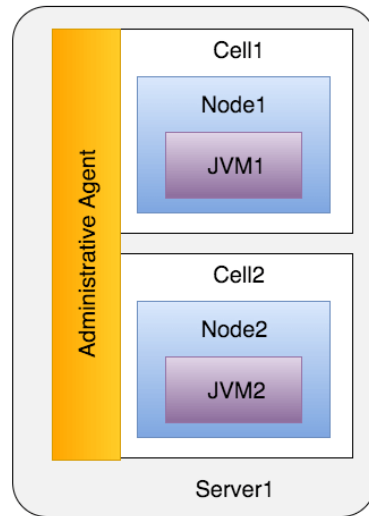


## Chapter 8

3. Provide examples of three types of middleware technology and a brief description of how they act as a bridge between the operating system and other applications in a network.

a **IBM WebSphere**

- i *Using the WebSphere Application Server, developers can connect users to Java applications called servlets that run on the server. The servlets serve as the middleman between the OS and the other applications. This allows administrators to manage specific compartments (nodes in this topology) bridging all the data to one processing point.*



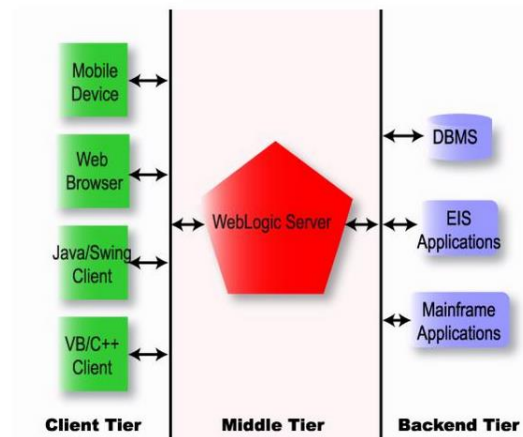
ii

**Administrative Agent Topology**

b **Oracle WebLogic**

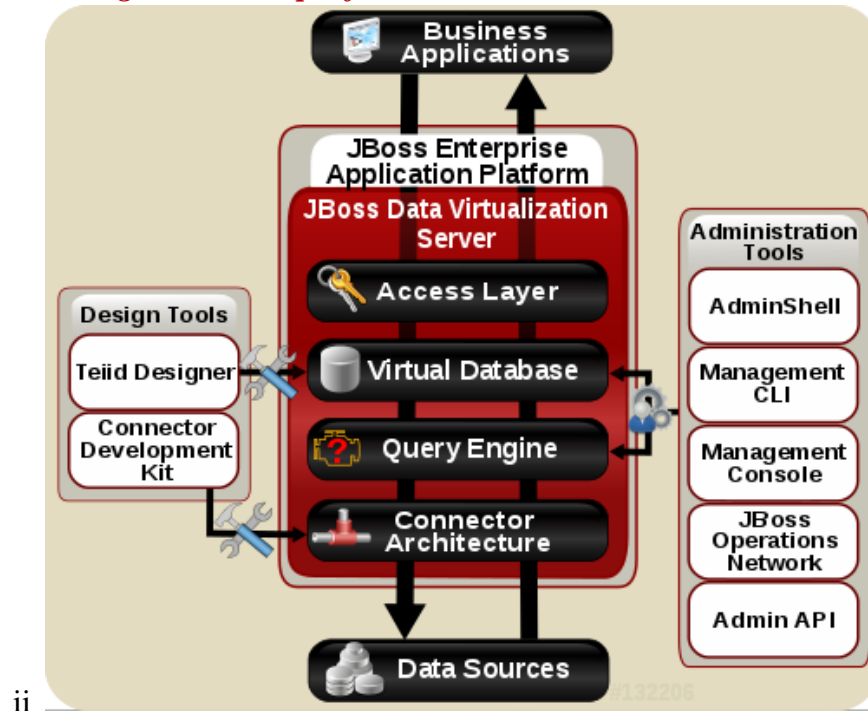
- i *Allows for the development, deployment, and running of enterprise application on premise and in the cloud. WebLogic manages system-level details to allow easy integration of corporate data and mainframe applications. It acts as the bridge that relays and processes the information between the client and backend side.*

Figure 1-1 Three-Tier Architecture



c **Red Hat JBoss Enterprise Application Platform (JBoss EAP)**

- i *Built to provide simplified deployment and full Jakarta EE performance for applications in any environment. JBoss EAP maximizes administrative productivity by making it easy to maintain and update your deployments. In addition, it features an updated management console user interface with navigation and support for large-scale domain configurations. Simply put, JBoss bridges business applications, administration tools, data sources, among others on a platform in the network.*



4. What is the difference between APIs and middleware?

- a **Application Programming Interface (APIs)** are used to interface with the software (or system) that a programmer is trying to interact with. APIs plays a direct role in providing a pipe for the one-to-one connection between you and to the tool, website, software, ...etc. you are interacting with. APIs does not impose a structure to the programmer as APIs do not necessarily have restriction on what you can do with them. They are simply a straightforward method of integration.
- b On the other, middleware adds an additional layer of complexity as it processes data from multiple sources to multiple destinations. Middleware incorporates layers of a system to be able to interact with one another, playing an active role in the actual cohesion of the final product. Middleware supplies a lot of tools in making different parts of the system able to integrate with one another. One downside of middleware is that it might possibly be competing or duplicating other functionalities of the system resulting in unnecessary work. Furthermore, middleware extensions are hard-coded, which adds restrictions to the structure

as you might have rewrite code if you want to update or transfer to a new platform. A usage of middleware would be in an E-commerce business in which you have a website and payment system, which you would use middleware to incorporate them together for the final payment process.

5. What are three different data formats for APIs

**The most common direct data formats for APIs are:**

- a *Extensive Markup Language (XML)*
- b *JavaScript Object Notation (JSON)*
- c *YAML Ain't Markup Language (YAML)*

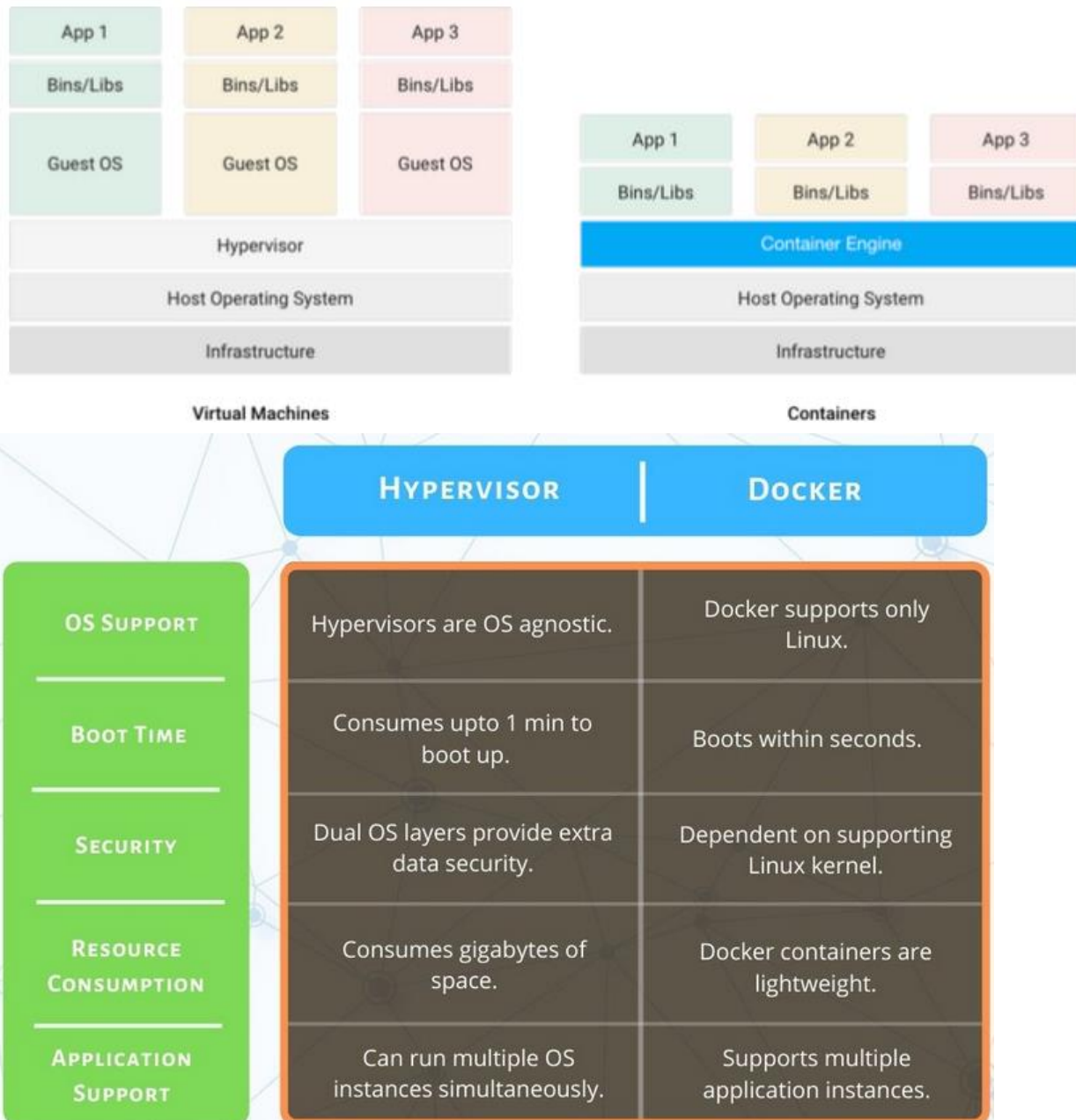
## Chapter 7

6. Describe the differences between type 2 hypervisors and the Docker Engine.

- a **Boot Time:** *Docker's boot time is significantly faster than hypervisor*
  - i Docker can create container within seconds, while a hypervisor must allocate more resources in booting up the OS as well as starting up any processes
- b **Resource Allocation:** *Hypervisors are more resource hungry than Docker containers*
  - i Hypervisors enable users to run a completely new OS on top of the currently working OS. On the other hand, docker does not boot a new OS, however, allows multiple instances of an application to run in different containers.
- c **Storage Usage:** *Hypervisors need more storage than Docker containers*
  - i Hypervisors consumes a lot of space on your drive as it installs the entire OS on your drive. Docker containers only store the image and all the necessary files that it needs. Simply put, you need more space with a hypervisor than a container.

**Overall, docker containers are lightweight and allows for rapid deployment of applications. Hypervisors are more resource hungry as they take up more storage and RAM from your system, however, they do provide the capability of having a completely new OS on top of your current OS. In addition, on a security aspect, a dual OS layer does provide extra security in comparison to a single kernel running separate containers. Simply put, hypervisors and docker containers both have their advantages and disadvantages that make them different from each other as each provide a specific use case scenario that allows each one to outshine the other.**





7. List existing available IAAS, PAAS, and SAAS technologies and explain why you think they might be useful than other technologies in the same category.

a **Infrastructure as a Service (IAAS)**

i **AWS EC2**

1 *AWS EC2 provides secure, resizable compute capacity in the cloud. In comparison to other technologies in the same field, AWS has paved a path for its name to be the #1 cloud provider. Amazon allows easy configuration with its online web service interface as well as gives deployment of OS. Overall, Amazon's EC2 is a great product with a 99.99% availability for each instance in every region they provide.*



**b Platform as a Service (PAAS)**

**i Heroku**

- 1 *Heroku allows developers to build, run, and operate application in the cloud. In addition, Heroku supports several languages including Python, Java, and PHP to just name a few. In comparison to other technologies in the same field, Heroku is useful in a sense that their company only focuses on app development and scaling the apps to actual production.*

**c Software as a Service (SAAS)**

**i Google Apps (Google Docs, Google Sheets, Google Drive, ...etc.)**

- 1 *Google is prevalent in creating technology that is convenient for the user. In this case, Google Apps is useful as it provides an alternative to software that you need to download and install such the Microsoft Office Suite. Google makes the functionality of the Microsoft Office Suite available over the internet by simply logging in with your online account. You can access your files online with no hassle as well as software is always up to date.*

