Use of Docker Compose Technology in a Synology NAS for the deployment of a Capital Flow Control Web System for a Livestock Association

Rafael Cerino Frías, José Ángel Jesús Magaña, José Manuel Gómez Zea, José Ney Garrido Vázquez, José Alejandro Hernández Cadena

National Technological Institute of Mexico / Technological Institute of Villahermosa. Systems and Computing Department. Highway Villahermosa - Frontera Km. 3.5 Ciudad Industrial Villahermosa, Tabasco, Mexico. C.P. 86010.

Abstract

NAS (Network Attached Storage) technology is very common today, and Synology technology stands out among them due to its constant innovation and updates that allow the user to easily manage the NAS, which is why most small businesses they choose it for the deployment of their systems, but it had a great weakness: it only allowed websites that were made in PHP, Node JS and Python to be uploaded to the server, but this was solved with the implementation of Docker. This article will show how the Synology NAS was configured for the deployment of the project Capital Flow Control Web System for a Livestock Association with the help of Docker Compose.

Keywords: Docker, Docker Container, NAS.

1. INTRODUCCIÓN

Data is an essential asset for businesses, without access to their data, businesses may not be able to provide their customers with the desired level of service. Poor customer service, lost sales, or team collaboration issues are all examples of what can happen when information isn't available.

Each of the above issues contributes to lost efficiency and possibly lost revenue if customers have to wait for a system fault to be fixed. Also, when it comes to data storage, small businesses face other storage-related needs such as: options are too expensive, servers are not easy to operate (requires specialized IT staff), and storage capacity. growth.

NAS systems are becoming more and more popular with small SMEs looking for a storage solution that is affordable and expandable. In this article, we will analyze the collaboration between two technologies; Docker and NAS Synology, who have teamed up to help developers minimize delivery time for modern applications. The deployment of the Capital Flow Control Web System project for a Livestock Association will be taken as a case study.

2. DOCKER AND NAS SYNOLOGY

Docker is an open-source platform for developing, shipping, and running applications. It allows you to separate applications from your infrastructure, so you can quickly deliver software. With Docker, you can manage your infrastructure the same way you manage your applications. By leveraging Docker methodologies to quickly ship, test, and deploy code, you can significantly reduce the delay between writing your code and running it in production (Docker, 2021).

NAS (Network Attached Storage) is a smart storage device connected to your home or office network. You can store all the files of your family and colleagues on the NAS, from important documents to valuable collections of photos, music and videos. Using a web browser or mobile apps, you can access files and use various services provided by the NAS over the Internet (Synology Inc., 2022).

The combination of Docker and NAS Synology technologies has been a success for the Synology company, offering the possibility of managing hundreds of containers created by developers from all over the world in DSM, eliminating the limitations that it had to only PHP, Node JS and Python.

3. DOCKER ON NAS SYGNOLOGY

Docker is a lightweight software modularization solution powered by Docker Inc. Leveraging this cutting-edge technology, Synology offers more packages for seamless deployment on a NAS device, such as Redmine, GitLab, Nginx, Apache, Node JS, or Java. Docker is a virtualization application that offers the possibility to manage thousands of containers created by developers from all over the world in DSM Synology NAS system. The popular built-in image repository, Docker Hub, lets you search for shared apps from other leading developers (Synology Inc., 2022).

With Docker on a NAS you gain flexibility by being able to deploy Docker containers at will and get the most out of your NAS. With Synology it is very easy to import and export containers that facilitates the backup and transfer of containers. In addition, Synology has a built-in Docker center registry that allows you to directly search its database for the image you need. Synology created an interface that allows the monitoring of the resources consumed by the containers with information such as the CPU that is being used and the memory.
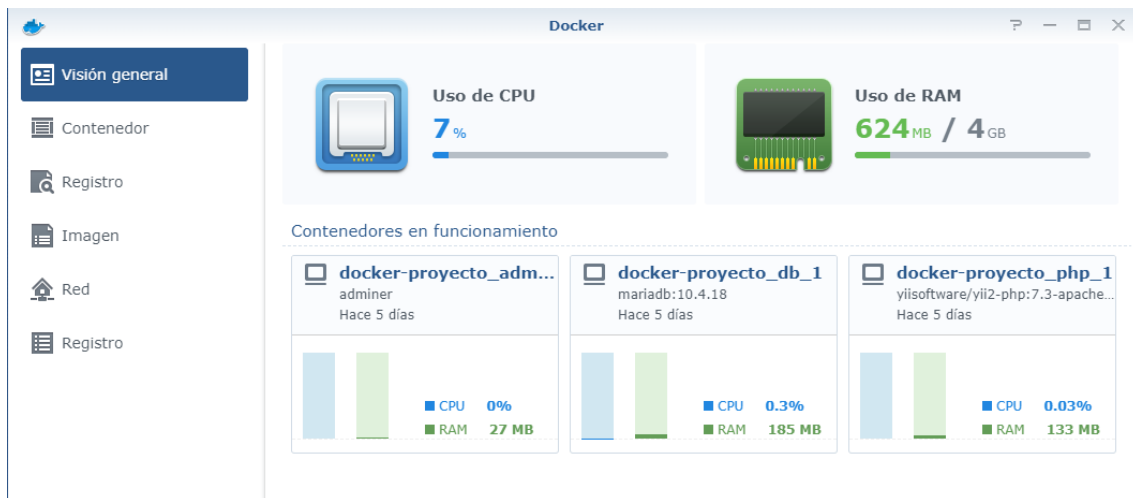

Figure 1. Docker interface on Synology NAS.

4. PROPOSAL

It is proposed to implement the Synology DS720+ NAS device for the deployment of the project Capital Flow Control Web System for a Livestock Association, having as justification that the company currently does not have a local server, nor a server in the cloud and neither does it have with a computer area that is capable of

doing the maintenance and administration of one. By taking possession of this device, you will get a 24-hour service, facilitating the problems that may arise when configuring the device.
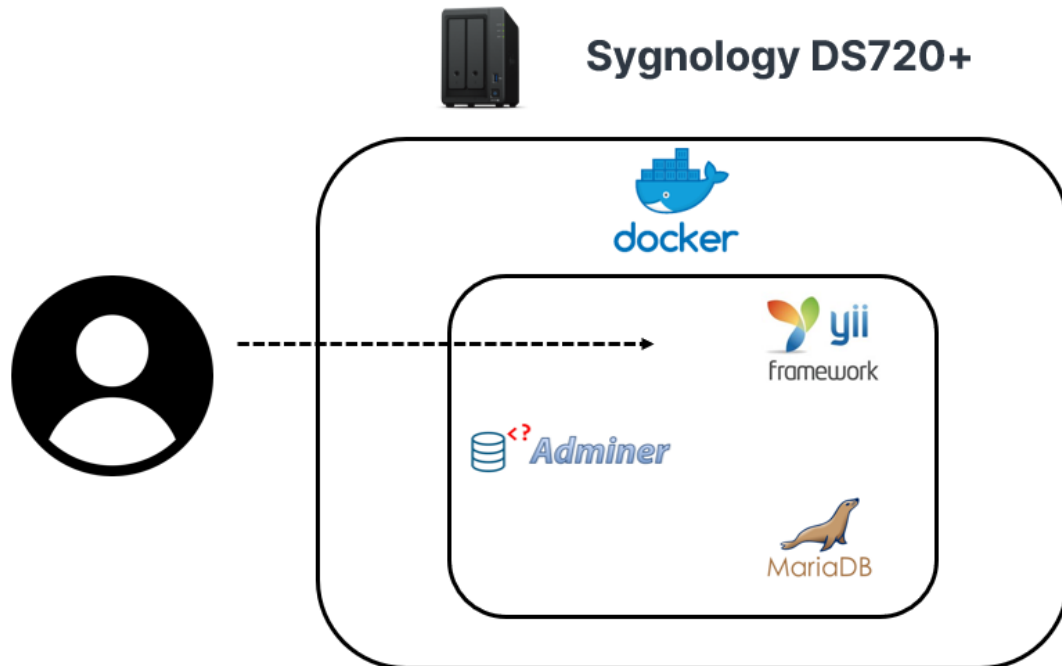

Figure 2. Project architecture.

In the previous Figure, the architecture of the project is shown; at the top layer we have Synology DS720+ as a NAS device, providing the tools for deployment; in the middle layer, Docker intervenes as a platform that manages the containers; finally, the Yii App, MariaDB and Adminer containers.

- **Yii App:** The container is made up of the Yii2 Framework and the NGINX web server.
- **MariaDB:** The container is made up of the MariaDB Database Management System (SGBD).
- **Adminer:** The container is made up of Adminer, the tool to manage database content graphically.

5. IMPLEMENTATION

Considering the problem and the proposed solution, the configuration of the NAS will be done after the initial configuration.

Installing Docker.

In order to start working with Docker we are going to go to the Synology package center, search for Docker and select install. Once installed two things will happen; All the packages related to Docker will be installed and a folder will be created where the necessary files can be uploaded to create the containers that are needed.

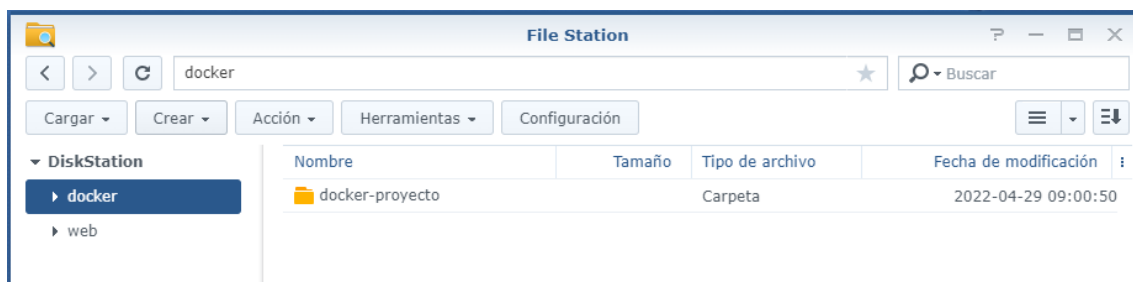**Figure 3.** Docker interface on Synology.

Figure 4. Docker folder.

Folder structure.

Since Docker Compose is going to be used for the project, it is necessary to place a structure that is based on what is needed:

- **mariadb_volume:** Folder where the project database is saved.
- **public_html:** Folder containing the project web program code.
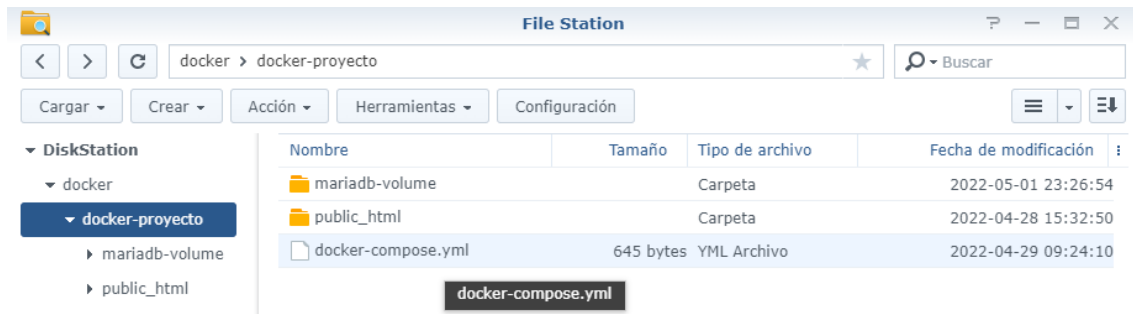- **docker-compose.yml:** File where the configuration that Docker will use to create the containers is specified.

Figure 5. Project folder structure with Docker.

Once the structure that is needed is created, we verify that the structure of the docker_compose.yml file is correct, as shown in the image below.

```
version: '3'
services:
  db:
    image: mariadb:10.4.18
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: example
    volumes:
      - ./mariadb-volume:/var/lib/mysql
    expose:
      - 3308
    ports:
      - 3308:3306
    networks:
      - nginxphp
  adminer:
    image: adminer
    restart: always
    ports:
      - 8088:8080
    depends_on:
      - 'db'
    networks:
      - nginxphp
  php:
    image: yiisoftware/yii2-php:7.3-apache-21.3.1
    restart: always
    volumes:
      - ./public_html:/app:delegated
    ports:
      - 8000:80
    depends_on:
      - 'db'
    networks:
      - nginxphp
networks:
  nginxphp:
```

Figure 6. Docker-compose.yml structure.

Permission settings.

To give permissions and allow Docker to Create, Edit and Delete files and folders for that we position ourselves in our folder and select properties, which will show us a tab that says permissions that will show us all the types of users that we have registered and we select the permissions write and read for the admin user.
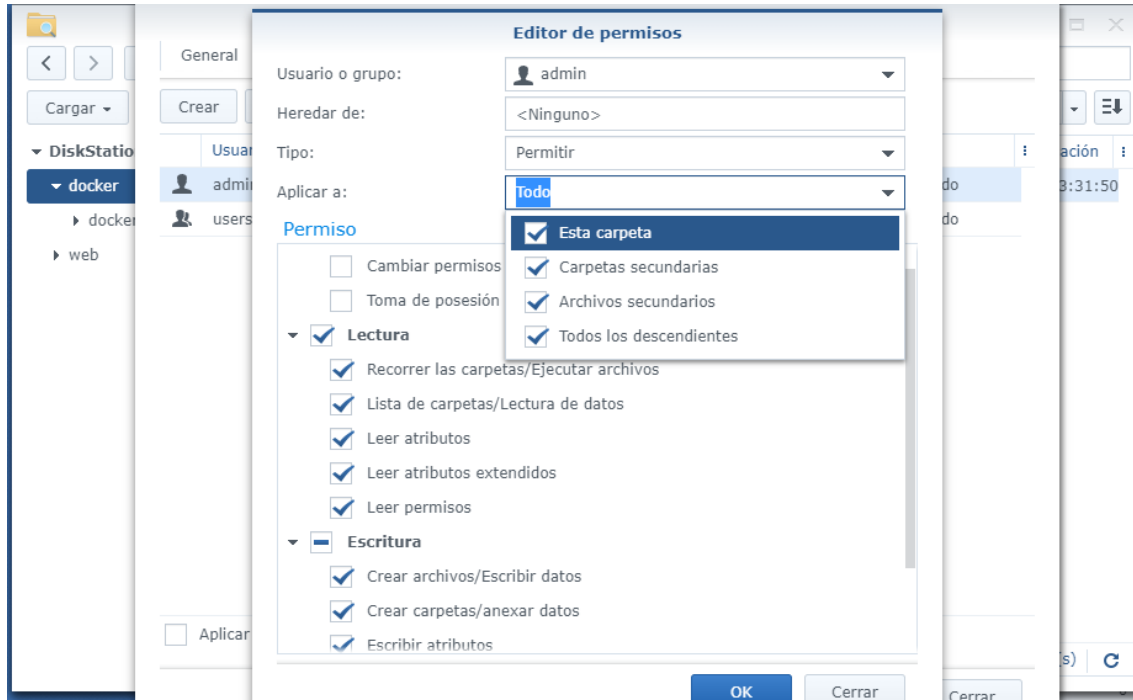
Figure 7. Permission settings.

Running Docker Compose.

In this part, the containers are created with the help of Docker Compose and the docker-compose.yml file that was configured previously, first you have to enter the NAS through the terminal and navigate to the location of the folder where we have our project. And finally, the sudo docker-compose up -d command must be executed, which will start the execution of our services with docker.

Figure 8. Docker Compose command.

6. **RESULTS**

When deploying the project Capital Flow Control Web System for a Livestock Association using the resources of the NAS device, a good performance is observed.

Figure 9. Docker main screen.

The Association can now easily manage the application, without worrying about the administration of a server and the large costs that it entails, it can now manage the NAS device and have 24/7 support from Synology.

In this case, which is a small Livestock Association that has 150 members, with a flow of operations between 1,200 and 1,500 monthly queries, up to 8 members per hour can be attended. So, a Synology DS720+ NAS is enough to meet these requirements. The Association would have to make an expense of 700 USD that already includes a domain and hard drives, which it recovers in 2 years without compromising its profits.

Figure 10. NAS Synology DS720+ and HDD.

7. CONCLUSION

Docker's addition to Synology's Packet Center put its NAS devices at the forefront of the competition in the SMB sector, especially for small ones, with competitive prices and services. It can be concluded that the implementation of the Synology DS720+ device was the correct decision taking into account the performance it shows, taking into account the cost and the lack of computing area.

REFERENCIAS

[1] Docker, "Docker es el futuro.," Diciembre 2021. [Online]. Available: https://docs.docker.com/get-docker/. [Accessed 11 December 2021].

[2] Synology Inc, "Synology.com," 2022. [Online]. Available: https://www.synology.com/es-mx/dsm/solution/what_is_nas. [Accessed 3 May 2022].

[3] Synology Inc, "Docker | Synology Inc.," 2022. [Online]. Available: https://www.synology.com/es-mx/dsm/feature/docker. [Accessed 3 May 2022].

Correo de autor: *rafacerino96@gmail.com*