АНОТАЦІЯ

Нанівський О.І., Рішняк І.В. (керівник). Інформаційна система розподіленого зберігання файлів. Бакалаврська кваліфікаційна робота. - Національний університет «Львівська політехніка», Львів, 2021.

Сьогодні існує багато розподілених файлових систем з різними архітектурами для різних потреб з своїми перевагами та недоліками, оскільки вони можуть значно спростити збереження і обробку великих даних. Переважно це громіздкі системи, які забезпечують високу продуктивність. Проте вони мають складну архітектуру та забирають багато часу і ресурсів для імплементації.

В цій роботі представлена розподілена файлова система з простою архітектурою, яка забезпечує легке встановлення і налаштування. В ній реалізовано процеси зберігання файлів, їх читання та маніпуляція простором імен. Дана система забезпечує прозорість використання, коли користувач не знає про архітектуру системи та спосіб зберігання файлів, а також її надмірність, щоб забезпечує автоматичне відновлення роботи системи та втрачених файлів.

Дана система надихалася архітектурними особливостями DFS, а також її аналогом з відкритим кодом HDFS [1,2]. Дані системи мають просту та ефективну архітектуру побудовану використовуючи досвід зберігання даних в Google та інших великих компаніях.

Побудову архітектури розподіленої файлової системи було зроблено за допомогою діаграми потоків даних, а також деталізовано процес зберігання файлу. Архітектура системи містить три основні елементи: клієнта, основний сервер та сервери даних. Основне навантаження потоку даних відбувається між клієнтом та серверами даних, основний сервер не бере участі в безпосередньому збережені даних, проте він відповідає за швидке збереження і отримання метаданих про файли та його блоки.

Дана система розроблялася і призначена працювати на операційній системі Linux. Це дозволить встановити програму на більшості віддалених серверів. Код програми написаний за допомогою мови загального призначення Python, що надає

усі інструменти для маніпулювання із операційною системою, а також віддаленого комунікування між клієнтом та віддаленими серверами. На стороні клієнта розроблений інструмент комунікації із системою - інтерфейс командного рядку, команди якого мають назву і схожий функціонал, як команди Linux.

Розроблена система забезпечує надійне зберігання даних за допомогою надмірності - репліки блоків і зберігання їх на різних серверах та прозорості - клієнт не знає про деталі внутрішньої архітектури системи і використовує файли як на локальній файловій системі. Також проста архітектура дозволяє легко та швидко розгортати систему.

Ключові слова - розподілена файлова система, DFS, HDFS, прозорість, надмірність, Linux.

Перелік використаних літературних джерел.

1. The Google File System [Електронний ресурс] / Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung Google – Режим доступу: https://storage.googleapis.com/pub-tools-public-publication-data/pdf/035fc972c796d33122033a0614bc94cff1527999.pdf .

2. Centralized Cache Management in HDFS [Електронний ресурс] / The Apache Software Foundation — Режим доступу: http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-hdfs/CentralizedCacheManagement.html .

ABSTRACT

Nanivsky O.I, Rishnyak I.V. Information system for distributed file storage. Bachelor's thesis. - Lviv Polytechnic National University, Lviv, 2021.

Today, there are many distributed file systems with different architectures for different needs with their advantages and disadvantages, as they can greatly simplify the storage and processing of large data. Mostly they are large systems that provide high performance. However, they have a complex architecture and take a lot of time and resources to implement.

This paper presents a distributed file system with a simple architecture that provides easy installation and configuration. It implements the processes of storing files, reading them and manipulating the namespaces. The system provides transparency of use, which means user is unaware of the system architecture and method of file storage, as well as its redundancy, to ensure automatic recovery of the system and lost files.

This system was inspired by the architectural features of DFS, as well as its open source analogue - HDFS [1,2]. These systems have a simple and efficient architecture which is built using the huge experience of Google and other large companies.

The architecture of the distributed file system was built using a data flow diagram, and the process of storing the file was detailized. The system architecture consist of three main elements: the client, the main server and the data servers. The main load of the data flow occurs between the client and the data servers, the main server is not involved in the data storing, but it is responsible for the rapid storage and retrieval of metadata about the files and its blocks.

This system was developed and designed to run on the Linux operating system. This will allow you to install the program on most remote servers. The program code is written in the general-purpose Python language, which provides all the tools for manipulating the operating system, as well as remote communication between the client and remote servers. On the client side, a tool for communication with the system has been developed - a command line interface, the commands of which have a name and similar functionality to Linux commands.

The developed system provides reliable storage of data by means of redundancy - replicas of blocks which are seved on different servers and transparency - the client does not know details about internal architecture of system and uses files as on local file system. Also, the simple architecture allows you to easily and quickly deploy the system.

Keywords: distributed file system, DFS, HDFS, transparency, redundancy, Linux.

References.

1. The Google File System [Електронний ресурс] / Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung Google – Режим доступу: https://storage.googleapis.com/pub-tools-public-publication-data/pdf/035fc972c796d33122033a0614bc94cff1527999.pdf .

2. Centralized Cache Management in HDFS [Електронний ресурс] / The Apache Software Foundation — Режим доступу: http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-hdfs/CentralizedCacheManagement.html .