Scholar Publishing Group
Distributed Processing System
https://doi.org/10.38007/DPS.2021.020206
ISSN 2790-0916 Vol. 2, Issue 2: 42-50

Distributed System Resource Management and Deployment Platform Considering Big Data

Amaren Charun*

The University of Sydney Business School, Australia

**corresponding author*

Keywords: Big Data, Distributed System, Resource Management and Deployment, Network Middleware

Abstract: With the development of the Internet and the advent of the mobile Internet era, to meet the business needs of big data, the scale of clusters is also increasing, and there are also problems in distributed automated development and automated management. The system is becoming more and more obvious. The purpose of this paper is to study the design and implementation of a resource management and distributed system development platform that considers big data. The platform is built on the OCF framework, and based on the existing OCF components, services are developed and deployed as components. The platform is based on the resource management of the underlying system, adopts an extensible node allocation strategy for node allocation, deploys and uninstalls services in the form of components, and closely integrates the development and deployment of services, thereby reducing costs and difficulty in operation and maintenance. Performance analysis test results show that when the data grows significantly, the time taken by the system also increases, especially when the data goes from 500 MB to 1 GB. The test results of service offloading latency show that data size has little effect on offloading service latency, but the overall efficiency of resource management and distributed system development platform is higher than that of Ansible.

1. Introduction

With the growing of network technologies such as the Internet, mobile Internet, and the Internet of Things, massive amounts of data are constantly being generated, and the world has entered the era of big data. Big data technology can effectively solve the problem that traditional stand-alone systems cannot store massive data and provide extremely high computing power [1]. The big data cloud platform has the characteristics of large scale, wide user range and many applications. The separate development of many big data computing frameworks is not only a waste of resources, but also complicated to operate and maintain. Therefore, how to schedule and manage various tasks and reasonably allocate cloud platform resources is of great significance to ensure the smooth operation. This paper mainly acquires a resource management and distributed system development platform, which can realize automatic cluster service deployment and automatic service management [2].

Scholars at home and abroad have done a lot of research on resource management and distributed system development platforms. The improved VRP optimization method proposed by Woiceshyn K has two main innovations: 1) using idle time as the objective function; 2) using a local search (LS) method. The idle time objective function is formulated to solve the isolation problem of the modified routing problem. On the other hand, the LS decision algorithm was developed to improve the efficiency of finding the optimal value of the chosen objective function. The proposed vehicle routing method has been validated with several simulated WiSAR scenarios, some of which have been included in this paper. The ability of the method to change the parameters of the problem is also investigated. The resource management problem addressed applies to situations where fleets visit a set of locations at predetermined times to provide services while performing other tasks in between. Such time-sharing applications include the development of sensor networks for on-site presence and rescue or fire monitoring, ambulance services that can handle emergencies, and courier services that can quickly process courier requests [3]. Awaysheh FM proposes a new taxonomy that classifies these models according to the communication system underlying them. We first discuss the horizontal questioning criteria that underlie the BD communication model and compare their main characteristics. The advantages/challenges of these domains compared to traditional local BD clusters are described. Since then, we have extensively studied modern BD development architectures and categorized them according to their underlying architecture. Finally, future research directions and suggestions for improving the status quo of BD are proposed for BD practitioners in emerging projects to support BD applications and big data analysis [4]. To sum up, some research achievements have been made in the design and implementation of distributed system resource management and deployment platforms at home and abroad, which is very inspiring and meaningful for the development of this research work.

This paper firstly studies the research status of related technologies at home and abroad, and compares the advantages and disadvantages of traditional distributed system resource management and deployment platforms. According to scientific research results, design and develop corresponding functional modules to realize resource management systems for different big data load applications, and provide effective resource management solutions for modern cloud computing data centers. For the heterogeneous load big data application in the cloud environment, a fine-grained and energy-efficient resource management mechanism is proposed to provide an effective resource management solution for modern cloud computing data centers. The research results of this paper will be applied to large-scale cloud computing data centers to provide a high-performance and low-cost support platform for various new big data applications such as data mining and data mining.

2. Research on Distributed System Resource Management and Deployment Platform Considering Big Data

2.1. Data Mining

Data mining refers to the discovery of hidden, unknown but useful information from massive, random, noisy, incomplete data of practical applications and knowledge processes. Databases based on data mining include many data stores, including relational databases [5-6]. Scientific manipulation of data using a variety of techniques. This technique can be used in many disciplines, including machine learning, databases, and statistics. Data mining allows people to extract data from data. Knowledge mining to support decision making [7-8].

2.2. Features of Distributed Systems

(1) Resource sharing

A resource here can be anything, such as printers, computers, disks, files, data, storage devices, websites, and networks [9-10].

(2) High system reliability

The distributed nature of a power distribution system makes it imply a kind of fault tolerance, if your work session makes the entire system at least partially available and working to a certain extent; while on one machine, when a computer fails, The entire system will not continue to work [11-12].

(3) Parallelism

Due to the limited processing power of computer systems, due to many factors (such as time difference), the availability of each computer is not equal. Distributed systems can make multiple computers in the same network work together, process in parallel, improve the efficiency of the whole system, use the load balancing algorithm to balance the load of each computer, and improve the performance of the whole system [13-14].

2.3. Network Middleware

In the field of computer networking, middleware (Middle Box) or network service (Network Service) is a network device that has the function of detecting, processing or controlling the flow of data, not just forwarding. Intrusion detection systems, firewalls, and load balancers are all common middleware. Intrusion detection systems and firewalls can provide network security, and load balancers can improve performance to prevent performance degradation due to load imbalance [15-16]. Today, middleware appliances such as firewalls, load balancers, and intrusion detection systems are widely used to distribute cloud data across multiple tenants. These intermediates are associated with application formats and shared by tenants for data transformation [17-18].

3. Design and Research of Distributed System Resource Management and Deployment Platform Considering Big Data

3.1. Platform Architecture Design

The resource management platform of this paper is divided into resource layer (Resource Layer), management layer (Management Layer), application layer (Application Layer) and presentation layer (Presentation Layer) from top to bottom, as shown in Figure 4.2. The resource layer is infrastructure resources, including physical host servers, storage and network resources; the management layer integrates some tools for monitoring and managing the system, such as the container orchestration framework Kubernetes; the application layer describes the services provided in the resource management platform, such as code warehouse Gitlab, warehouse manager Nexus, identity authentication LDAP, to complete the management of resources such as code, mirroring and personnel, as well as some heavy computing tasks (such as deep learning) related services; the presentation layer is the user operation and use management platform The user can access the services provided by the backend through the browser, and perform operations such as remote debugging, service management and visual resource monitoring. Services run on the platform as containers and are managed by Kubernetes.

3.2. Platform Function Module Design

In the platform development module, the platform development module must meet various requirements such as user login, visitor login, creation of development cluster foundation, core cluster service selection, big data component service selection, development progress view, support components, etc. Customized installation and advanced installation, Powerful kernel and maintenance-friendly development interface, etc. In the platform management module, the platform management module must meet the monitoring management, user interface and other requirements of the core work and development. The platform function module is shown in the figure below:

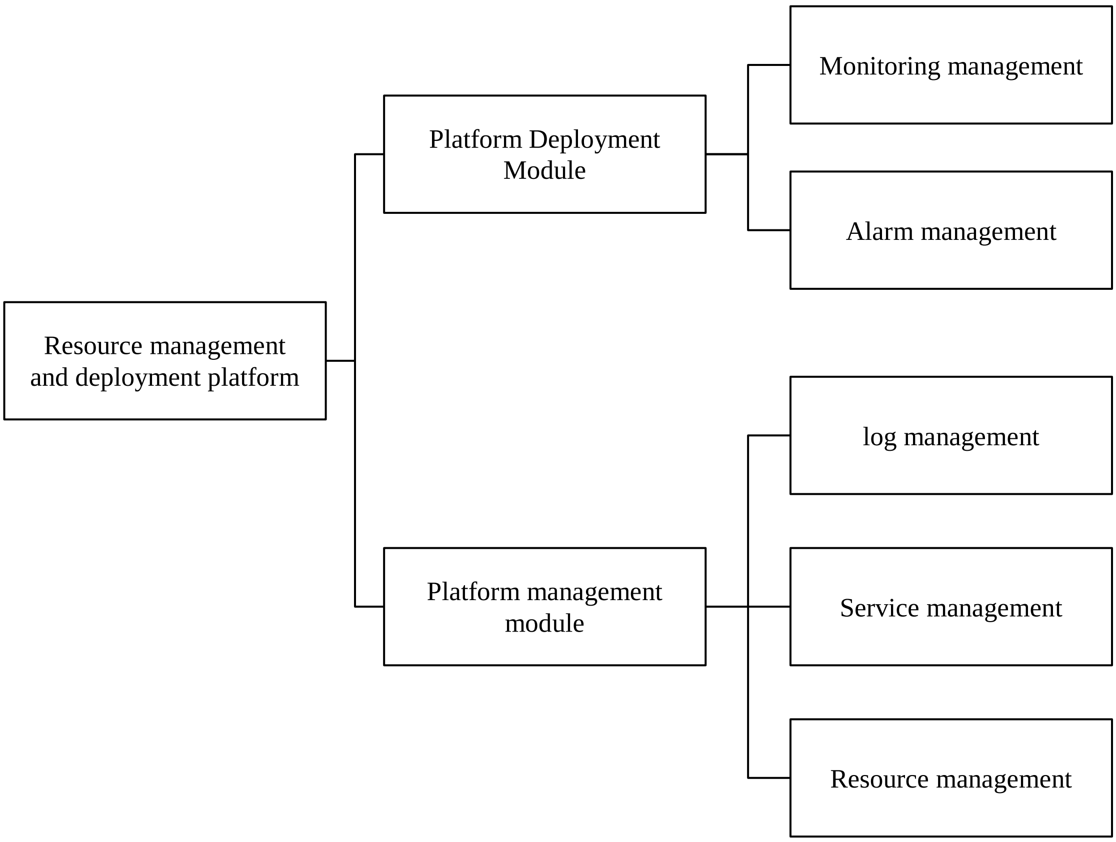

Figure 1. Functional module design of resource management and deployment platform

As shown in the use case of the platform development module in Figure 1, the platform development function is responsible for four functions: host management, development platform management, cluster development, and project management. When using the platform management module, find fault conditions and manage notification alarms.

3.3. Analysis and Solution of Data Center Resource Planning Model

On the basis of using queuing theory to model the data center resource planning problem, the data center resource planning model is analyzed by using the birth and death model of stochastic process, and the dynamic arrival and server status of big data application virtual machine requests in the system can be analyzed. Transfer is constructed as a birth-and-death process. Define J(t) as the number of virtual machines running in the system at time t, and L(t) as the number of servers in the open state at time t, then the random state transition space of both J(t) and L(t) is:

$$\Omega = \{(i,j): 0 \le j \le M-1, 0 \le i \le j\} \cup \{(i,j): j \ge M, 0 \le i \le N\} \tag{1}$$

Let W(t) be the probability when the virtual machine arrives at time and runs before time less than t. If there are i servers at time t, then:

$$W_i(0) = \begin{cases} 0 & i = 0 \\ y_i^{-1} \sum_{p=0}^{i-1} \pi_{ip} & 1 \le i \le N \end{cases} \tag{2}$$

The physical meaning of formula (2) is that when a virtual machine request arrives, if there is no running server, the request will be rejected directly. Otherwise, the virtual machine will wait for the running task to finish releasing resources.

4. Platform Test Results and Analysis

4.1. Delayed Testing of Service Deployment

Data file sizes are 0, 50 MB, 100 MB, 500 MB, and 1 GB. Four datasets are included with different data file sizes. The running delay of Ansible in this system and test is known in Table 1, and the average delay is shown in Figure 2.

Table 1. Latency test results for service deployment

Data size	0	50MB	100MB	500MB	1GB
Ansible(s)	6.1	7.8	15.3	42.31	90.25
	6.4	7.6	15.2	43.21	92.36
	6.2	6.9	15.4	45.03	93.45
	5.9	7.2	14.9	43.36	89.67
Deploy Platform(s)	0.19	0.64	5.26	23.62	54.35
	0.17	0.68	5.43	24.35	53.45
	0.16	0.62	5.37	21.36	55.21
	0.16	0.69	5.18	23.45	54.36

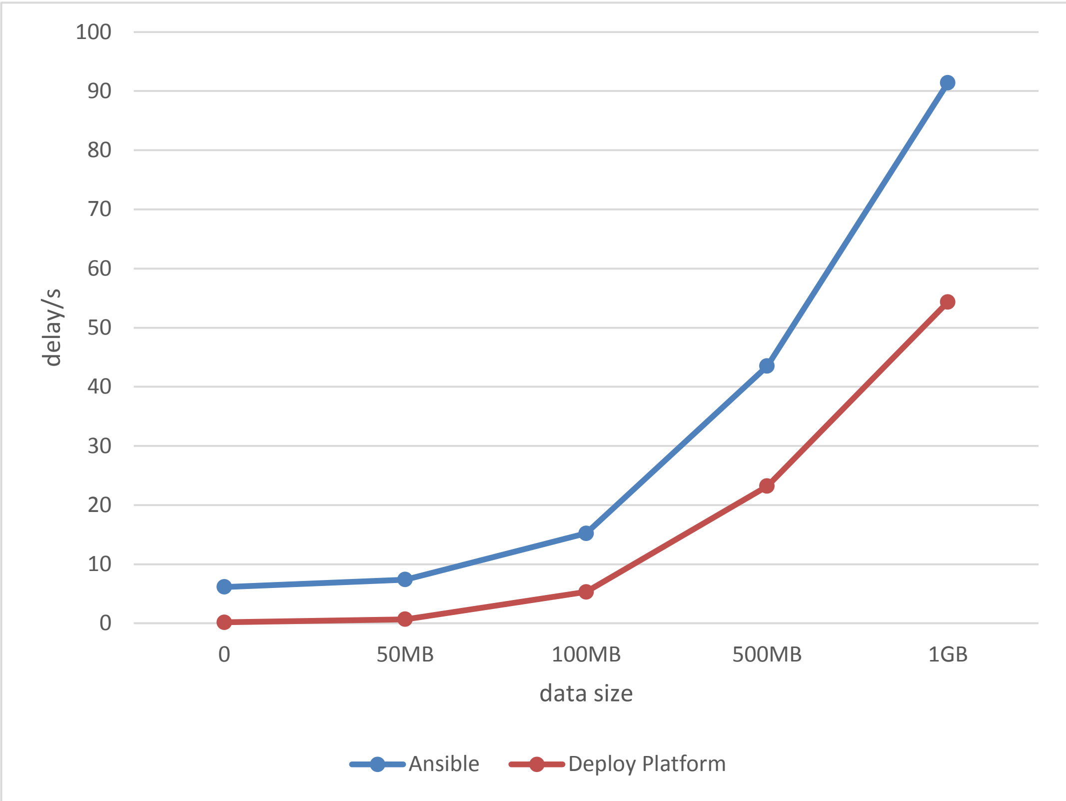

Figure 2. Service deployment average latency test results as data size changes

As shown in Figure 2, when the data increases significantly, the time spent by the system also increases, especially when the data changes from 500MB to 1GB.

4.2. Delay Test of Service Unloading

After each deployment is completed, we will also perform the corresponding uninstallation tasks, so that the subsequent deployment tasks can run smoothly, and count the delay of each uninstallation task. Here, 4 sets of data are selected, as shown in Table 2. The average time delay is shown in Figure 3.

Table 2. Latency test results for service offload

Data size	0	50M	100M	500M	1G
Ansible(s)	4.2	3.7	3.6	4.2	4.6
	3.2	3.5	3.8	4.1	4.5
	3.3	2.9	3.4	3.8	4.8
	3.2	3.8	3.9	3.9	4.7
Deploy Platform(s)	0.052	0.053	0.062	0.089	0.112
	0.044	0.047	0.069	0.094	0.154
	0.046	0.062	0.068	0.087	0.167
	0.045	0.055	0.067	0.091	0.128

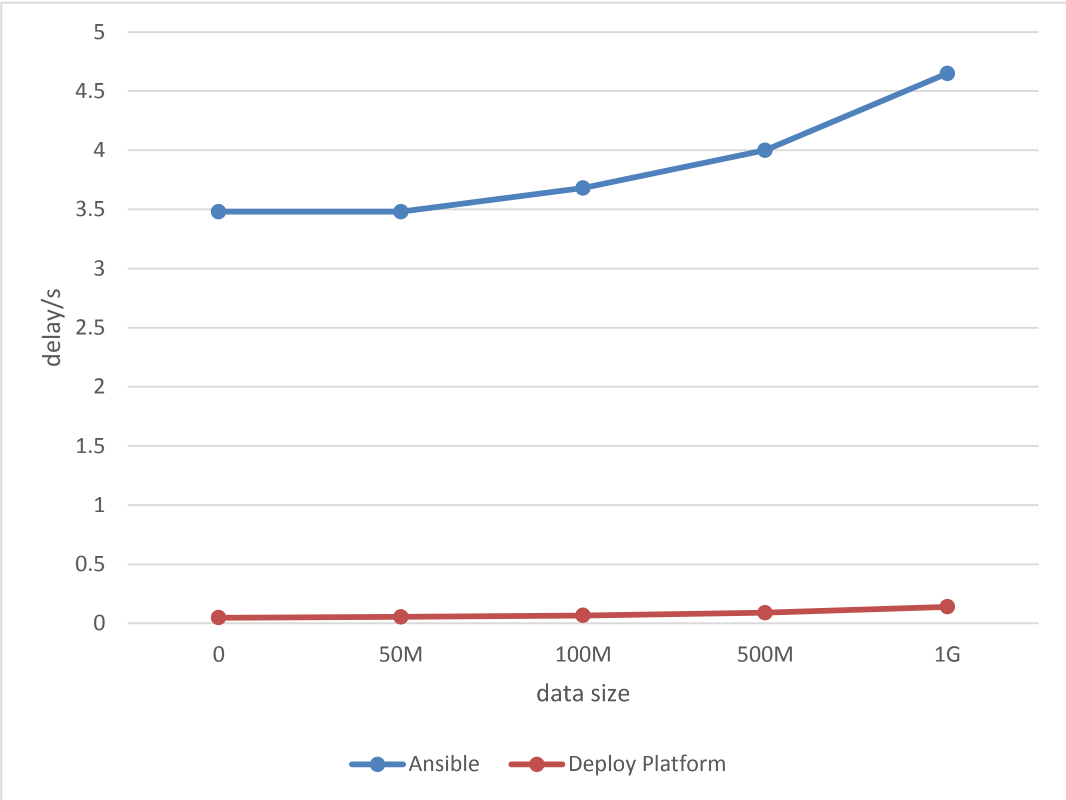

Figure 3. Service offload average latency test results as data size changes

As can be seen from Figure 3, the size of the data does not have much impact on the latency of offloading services, but the overall efficiency of the resource management and distributed system development platform is higher than that of Ansible. In general, resource management and distributed development platforms perform well in terms of latency performance, so distributed system resource management and deployment platforms have good performance while fulfilling their own functional requirements.

5. Conclusion

The platform provides big data producers with unique big data capabilities such as collecting, storing, processing and accessing large amounts of disparate data from multiple sources. Based on the analysis of dividing data blocks by random samples, the statistical results of large data sets can be estimated without calculating the entire data set. Therefore, after the large data sets are converted into random sample division expressions, the size of big data is no longer big data. Barriers to analysis. This increases the power of cluster computing when the amount of data exceeds the available resources. A distributed partitioning algorithm based on a random sample partitioning data representation model proposed in this paper provides certain technical support for analyzing large data sets across data centers, but requires further development of detailed techniques to facilitate the analysis of large data sets across multiple data centers. Analyze large data sets.

Funding

This article is not supported by any foundation.

Data Availability

Data sharing is not applicable to this article as no new data were created or analysed in this study.

Conflict of Interest

The author states that this article has no conflict of interest.

References

[1] Papageorgiou A, Fernandez-Fernandez A, Siddiqui S, et al. On 5G network slice modelling: Service-, resource-, or deployment-driven?. Computer Communications, 2020, 149(Jan.):232-240. https://doi.org/10.1016/j.comcom.2019.10.024

[2] Ray L. Enterprise Resource Planning Deployment and Business Process Reengineering at a Major University: A Case Study. International journal of strategic information technology and applications, 2018, 9(1):98-109. https://doi.org/10.4018/IJSITA.2018010105

[3] Woiceshyn K, Kashino Z, Nejat G, et al. Vehicle Routing for Resource Management in Time-Phased Deployment of Sensor Networks. IEEE Transactions on Automation Science and Engineering, 2018, PP(2):1-13. https://doi.org/10.1109/TASE.2018.2857630

[4] Awaysheh F M, Alazab M, Garg S, et al. Big Data Resource Management & Networks: Taxonomy, Survey, and Future Directions. IEEE Communications Surveys & Tutorials, 2021, PP(99):1-1.

[5] Yoon S, Cho J H, Dan D K, et al. DESOLATER: Deep Reinforcement Learning-based Resource Allocation and Moving Target Defense Deployment Framework. IEEE Access, 2021, PP(99):1-1.

[6] Elsahn Z, Siedlok F. "Can we build it? Yes, we can!" complexities of resource re-deployment to fight pandemic. Industrial Marketing Management, 2021, 93(3):191-207.

[7] Busolo G, Nderu L, Ogada K, et al. Application of a Multilevel Technology Acceptance Management Model for Effective Technology Deployment. International Journal of Computer Science and Information Technology, 2021, 13(1):57-74. https://doi.org/10.5121/ijcsit.2021.13105

[8] Thurasamy R, Alfarraj O, Alzahrani A, et al. End users' resistance behaviour paradigm in pre-deployment stage of ERP systems: evidence from Bangladeshi manufacturing industry. Business Process Management Journal, 2021, 27(5):1496-1521. https://doi.org/10.1108/BPMJ-08-2019-0350

[9] Rahman A, Islam M J, Montieri A, et al. SmartBlock-SDN: An Optimized Blockchain-SDN Framework for Resource Management in IoT. IEEE Access, 2021, PP(99):1-1.

[10] Ahmad T, Singh S P, Johri P, et al. Cloud computing for mobile applications: outsourcing the deployment of virtual machines. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 2021, 12(14):2120-2132.

[11] Avasalcai C, Tsigkanos C, Dustdar S. Resource Management for Latency-Sensitive IoT Applications with Satisfiability. IEEE Transactions on Services Computing, 2021, PP(99):1-1.

https://doi.org/10.1109/TSC.2021.3074188

[12] Keshavamurthy P, Pateromichelakis E, Dahlhaus D, et al. Edge Cloud-Enabled Radio Resource Management for Co-Operative Automated Driving. IEEE Journal on Selected Areas in Communications, 2020, PP(99):1-1.

[13] Kazakova N, Mel'Nik M, Dudorova E. Prospects for Implementing Big Data Analytics into the Auditing Profession. Auditor, 2021, 7(3):40-47.

[14] Zacharias J. Addressing Global Climate Change With Big Data-Driven Urban Planning Policy. International Journal of E-Planning Research, 2021, 10(It 4):1-16. https://doi.org/10.4018/IJEPR.20211001.oa1

[15] Lorenc A, Burinskiene A. Improve the orders picking in e-commerce by using WMS data and BigData analysis. FME Transactions, 2021, 49(1):233-243. https://doi.org/10.5937/fme2101233L

[16] Laterza V. Could Cambridge Analytica Have Delivered Donald Trump's 2016 Presidential Victory? An Anthropologist's Look at Big Data and Political Campaigning. Public Anthropologist, 2021, 3(1):119-147.

[17] Mohamed M A, Abdel-Fattah M A, Khedr A E. Challenges and Recommendations in Big Data Indexing Strategies. International Journal of e-Collaboration, 2021, 14(2):22-39. https://doi.org/10.4018/IJeC.2021040102

[18] Awan M J, Shafry M, Rahim M, et al. Social Media and Stock Market Prediction: A Big Data Approach. Computers, Materials and Continua, 2021, 67(2):2569–2583. https://doi.org/10.32604/cmc.2021.014253