Research on Computer Basic Practice Course Based on Huawei Cloud Platform

Yu Liu, Jizhao Cui, Yi Zhou, Ruimei Gao*
Chengxian College, Southeast University, Nanjing, Jiangsu, China
**Corresponding Author*

Abstract: In order to solve the problems of insufficient hardware resources, tedious deployment of experimental environment, and difficulties in cross-regional cooperation in traditional computer experiment teaching, this paper proposes a solution to build a computer experiment platform and site based on enterprise cloud technology. The solution takes the enterprise cloud host as the core hardware support, combines the enterprise cloud elastic computing, cloud storage, security protection and other services, and builds a scalable and highly reliable experimental site. Firstly, according to the requirements of computer experiment teaching (such as program design, network configuration, database development, etc.), the ECS elastic cloud server configuration of enterprise cloud was selected to determine the CPU, memory, storage and operating system version. Secondly, the enterprise cloud console was used to complete the initialization of the cloud host, the security group rules were configured to open the required ports of the experiment, and the experimental resources and student achievements were stored through the Cloud Storage Service (OBS). Thirdly, the experimental environment was deployed on the cloud host, including the installation of programming language compilers (Java, Python), and the experimental site was built based on Web technology. Finally, the performance of the platform was verified through actual testing.

Keywords: Enterprise Cloud; Computer Experiment Platform; Elastic Cloud Host; Experiment Site; Cloud Storage

1. Introduction

In computer experiment teaching, the traditional laboratory mode based on local hardware faces multiple challenges. On the one hand, the computer hardware updates quickly and the cost of laboratory equipment procurement and maintenance is high, which is difficult to meet the needs of artificial intelligence, big data and other emerging experiments for high-performance hardware. On the other hand, the deployment of the experimental environment depends on the local server, and different experiments (such as network protocol analysis and distributed system development) need to configure the environment separately, which is time-consuming and prone to version conflicts. In addition, in special scenarios such as the epidemic, students cannot enter the offline laboratory, and it is difficult to guarantee the continuity of cross-regional experimental collaboration and learning.

With the development of cloud computing technology, cloud platform has become an important direction to solve the pain points of traditional experimental teaching with the advantages of elastic expansion, on-demand allocation, remote access and so on. Enterprise cloud is a leading cloud computing service provider in China. Its services such as Elastic Cloud Host (ECS), Object Storage (OBS) and Cloud Container Engine (CCE) have high availability, high security and strong compatibility, which can meet the diverse needs of computer experimental teaching. [1] At present, some universities have tried to introduce cloud computing technology into experimental teaching, but most of them focus on a single experimental scenario (such as program design), lack of adaptation of multiple types of experiments and integrated site construction, and do not fully combine the technical characteristics of enterprise cloud for deep optimization.

Based on this, this paper proposes a complete scheme to build a computer experimental platform and site based on enterprise cloud technology, which takes the enterprise cloud host as the core, integrates multiple types of cloud services to build an experimental environment, develops a fully functional experimental site, and realizes the whole process of experimental resource management, environment deployment, task interaction and achievement evaluation online. The feasibility and effectiveness of the scheme are verified by teaching practice. This paper provides a reference for the digital transformation of computer experimental teaching. The front-end is based on Vue.js to achieve a responsive interactive interface, the back-end is built with Spring Boot to build a microservice system, and the experimental container cluster is deployed in combination with the enterprise cloud ECS. Docker and Kubernetes are used to achieve environment isolation and rapid orchestration. [2] The experimental data is stored in OBS.

Fine-grained access control was implemented through IAM to ensure the security of concurrent access of multiple users. The platform supports automatic scheduling of computing power resources according to experimental requirements, realizes GPU/CPU elastic scaling, and meets the differentiated load requirements of courses such as deep learning and compilation principles. The whole process of the experiment can be traced back, and the logs and results are automatically archived to OBS and generated analysis reports to help optimize teaching evaluation. The actual teaching test on October 31, 2025 showed that the scheme significantly improved the efficiency of experimental preparation and cross-regional collaboration ability, and the resource utilization rate was 68% higher than that of the traditional mode. Students can seamlessly access the experimental environment through the Web end to realize anywhere, anytime learning. Teachers can dynamically adjust teaching strategies based on platform data to improve their personalized guidance ability. In the future, we will explore the introduction of high-quality computing power evaluation system, optimize resource scheduling algorithm, and continue to improve the level of computational efficiency and intellectual efficiency to help build a green, efficient and sustainable digital experimental ecology.

2. The Overall Architecture Design of the Experimental Platform Based on Enterprise Cloud

2.1 Architecture Design Principles

The experimental platform architecture design follows the following principles:

1) Elastic expansion, according to the number of experiment and load dynamic adjustment of cloud host configuration, to avoid resource idle or overload;

2) Secure and reliable, through the enterprise cloud security group, data backup, access control and other mechanisms to ensure the safety of experimental data and environment;

3) Ease of use, simplifying the deployment process of experimental environment, and reducing the operation threshold of teachers and students;

4) Compatibility, supporting multiple types of computer experiments (programming, network, database, etc.), adapting to different operating systems and development tools.

5) Maintainability: microservice architecture is used to decouple modules, which facilitates function iteration and troubleshooting.

Each component is uniformly scheduled through the API gateway, and the log is collected to the enterprise cloud LTS service to improve the operation and maintenance efficiency. Relying on the global node layout of enterprise cloud, the platform supports cross-regional experimental resource scheduling to ensure service stability in high concurrent scenarios. In the teaching practice, teachers can preset the experiment template and deploy the environment with one click. Students can obtain an independent sandbox environment through the browser to complete the whole process of code writing, compiling and running and result submission. All operations were recorded in real time, and AI-assisted scoring model was combined to achieve automatic scoring of objective questions and procedural evaluation of subjective questions.

2.2 Overall Architecture

The overall architecture of computer experiment platform based on enterprise cloud is divided into three layers, which are infrastructure layer, experimental environment layer and application site layer. [3-5] The corresponding relationship between the functions of each layer and enterprise cloud services is as follows:

The infrastructure layer relies on IaaS services of the enterprise cloud to provide computing, storage, network and other basic resources, including elastic cloud server ECS, object storage OBS, virtual private cloud VPC and bandwidth resources, to support the flexible construction and high-availability deployment of the upper experimental environment. The experimental environment layer implements resource scheduling and isolation based on containerization technology and Kubernetes cluster, combines image warehouse SWR to encapsulate various experimental scene templates, supports high-performance demand scenarios such as GPU acceleration and distributed training, and automatically deploys and recovers experimental instances through DevOps pipeline. At the application site level, Web-based interactive interface is provided to teachers and students, which integrates functional modules such as experiment management, online editing, real-time collaboration, and intelligent evaluation. The educational administration system and LTS log service are connected through API gateway to realize unified identity authentication, operation audit, and visual analysis of teaching data.

2.2.1 Infrastructure layer

This layer provides hardware and basic service support for the platform, and the ECS elastic cloud host is used as the experimental running carrier. According to the experimental requirements, the ECS instance of "general computing type" is selected, which is configured with 2 cores of 4G memory (basic experiment) or 8 cores of 16G memory (high performance experiment, such as big data processing), and the combination of "cloud disk + OBS" is used for storage: Cloud hard disk is used to install operating system and experimental software, and 40GB high-performance cloud hard disk is selected to ensure the running speed. OBS is used to store non-real-time access resources such as experimental instructions, courseware, and experimental reports submitted by students. It supports massive data storage and convenient rights management. At the network level, an isolated private network environment was constructed through VPC, and a security group policy was configured to limit port access to ensure the safety of experimental data transmission. Bandwidth resources are allocated on demand, and elastic scaling is supported to cope with high concurrent experimental scenarios. All ECS instances are deployed across availability zones, combined with the automatic failover mechanism of the cloud platform to ensure service continuity and high resource availability. The ECS layer connects to the upper monitoring system through the enterprise cloud unified operation and maintenance interface, and collects performance indicators such as CPU, memory, and disk IO in real time to provide data support for resource optimization.

In addition, the infrastructure layer also integrates enterprise cloud security services, including security group (control cloud host inbound/outbound traffic, open SSH, RDP, HTTP and other necessary experimental ports), cloud firewall (resist external network attacks), data backup service (regularly backup ECS instances and OBS data to prevent data loss). To ensure the stable operation of the experimental platform. In terms of network, the virtual private cloud VPC was used to achieve resource isolation to ensure that the networks between different experimental environments did not interfere with each other. At the same time, elastic public network IP and NAT gateways were configured to meet the external access requirements of the experimental instances. [6,7] All the infrastructure is deployed in the four regions of enterprise cloud North China and Beijing, and its high availability is used to ensure the continuous service of the platform. Combined with cloud monitoring services, the resource utilization rate and system health status are monitored in real time, timely warning and automatic expansion are provided to adapt to the fluctuation of resource demand during the peak period of teaching.

2.2.2 Experimental environment layer

Based on the ECS instances of the infrastructure layer, this layer constructs a standardized environment suitable for different types of computer experiments. Through the enterprise cloud "mirror service" to create the experiment environment image, the operating system (Windows Server 2019 or CentOS 7), experimental software (such as Java JDK 1.8, Python 3.9, MySQL 8.0, Docker 20.10, Wireshark 4.0) and configuration parameters (e.g., environment variables, software dependencies) are pre-installed into the image. When experiments need to be carried out, ECS instances can be quickly created directly based on the image, and there is no need to repeat the deployment of the environment, which greatly reduces the preparation time. At the same time, special image templates were customized for different course requirements, such as programming experiment images preloaded with

IDE and compilation environment, network security experiment images integrated with Kali Linux tool set, and database course images configured with cluster mode and test data sets. For distributed experiments (such as distributed database and cluster deployment), a virtual network environment was constructed through the enterprise cloud "Private Network (VPC)". Multiple ECS instances were divided into the same VPC subnet, and the internal network IP was configured to realize the interconnection between instances, simulating the real distributed architecture. At the same time, the Load Balancing (ELB) service of the enterprise cloud was used to distribute requests to different ECS instances in high concurrent experimental scenarios (such as multiple students accessing the experimental environment at the same time) to avoid single point of overload. Through the elastic scaling strategy, the number of ECS instances is automatically adjusted by combining with cloud monitoring indicators to ensure efficient resource utilization and cost control. The life cycle of the experimental environment is managed by the platform, which supports the automatic creation and release of instances according to the course schedule to avoid resource idleness. All operations can be integrated into the teaching management system through the API, which realizes one-click experimental environment scheduling and status tracking, and comprehensively improves the intelligence and automation level of experimental teaching.

2.2.3 Application site layer

The experimental site is developed based on Web technology and deployed in a separate enterprise cloud ECS instance (2 cores with 2G memory to meet the requirements of lightweight Web services). The front-end uses Vue.js framework to build pages, the back-end uses Spring Boot framework to develop interfaces, and the enterprise cloud Relational Database Service (RDS) (MySQL 8.0) is used as the database to store structured data such as experimental tasks.

The core functions of the experimental site include: Experiment task management (teachers release experiment topic, instruction, deadline, support online editing and modification), distribution of experimental environment, students through the site application experiment ECS instance, automatically assigned account password and access the address), experimental results submitted (students upload experiment report, code file to the site, Support for online preview and version management), automatic scoring (for programming experiments, the test system is integrated, and the correctness of the code is automatically detected and scored through the preset test cases), and the whole process of experimental teaching is realized online. [8,9] At the same time, the site supports multi-role access control, and teachers can view students 'experimental progress and operation log in real time for procedural evaluation. Students can check the feedback results and grading rules at any time. By integrating enterprise cloud Model Arts services and introducing AI auxiliary functions, such as intelligent duplicate checking of experimental code, abnormal behavior recognition and learning path recommendation, the accuracy and personalization level of teaching management are further improved. PC interface responsive design adaptation and mobile terminal, security device access experience smooth.

3 The Specific Implementation of the Experimental Platform and Site

3.1 Initial Configuration of Enterprise Cloud Resources

Firstly, resource planning and VPC network configuration were completed in the enterprise cloud console, subnets were divided and security group rules were set to ensure safe and controllable communication between ECS, RDS, ELB and other services. Special teaching accounts were created through IAM services, and roles were assigned according to the principle of least privilege to ensure operation compliance. The cloud server backup strategy was used to mirror the key ECS instances to improve the system recoverability. All resources are labeled (e.g., "course number", "experiment type") to facilitate cost sharing and automated management. Then, the initialization script was written through the enterprise cloud SDK, and the ECS instance creation, RDS database instance deployment and network configuration were automatically completed, which greatly shortened the platform construction cycle. The Object Storage Service (OBS) was used to establish the experimental database to realize the unified storage and efficient distribution of teaching resources. Combined with the cloud monitoring service to collect the running status of each component in real time, the response time of abnormal alarm is less than 5 minutes to ensure the high availability of the platform. All the operation logs were connected to the cloud audit service to meet the traceability requirements of the teaching process. After the resource initialization is completed, the platform application layer deployment phase is entered.

3.1.1 ECS instance creation and configuration

Log in to the enterprise cloud console, go to the "Elastic Cloud Server ECS" page, and follow the steps below to create an ECS instance for experimentation:

(1) Select the instance specification: "g6.large.2" (2 core 4G) is selected for the basic experiment, and "c7.2xlarge.2" (8 core 16G) is selected for the high-performance experiment;

(2) Select the image: based on the public image of the enterprise cloud (Windows Server 2019 or CentOS 7), or use the custom experiment environment image (experiment software has been pre-installed).

(3) Configuration store: system disk choose 40 gb high-performance cloud drive, according to the experiment data plate needs to add (such as the big data experiment add 100 gb of data plate);

(4) Network configuration: add a custom VPC subnet, select "charged according to traffic" elastic public IP, to facilitate students to remote access;

(5) Set up a security group: create a "experimental platform security group", open ports include 22 (Linux SSH access), 3389 (Windows RDP access), 80 (Web services), 3306 (MySQL database), limit the source IP to the school campus network or designated network segment, improve security;

(6) Set the login credentials: Linux instance USES the key on the login, Windows instance using password, certificate by the experiment site of unified management and distribution.

3.1.2 OBS and RDS configurations

(1) OBS configuration: Create "experimental resource bucket" and "student achievement bucket", which are used to store public resources such as experimental instructions, courseware, and experimental reports and codes submitted by students respectively. Different access rights were set for the two buckets. The "experimental resource bucket" was set to "public reading" to facilitate students to download. The "student achievement bucket" was set as "private", only students and teachers could access it, and the bucket permissions were assigned through the enterprise cloud IAM (identity and access management) service [10].

(2) RDS Configuration: Create a MySQL 8.0 instance, select the "Universal type" specification (2 core 4G memory, 50GB storage), and configure the VPC to be the same as the ECS instance of the experimental site to ensure that the site has access to the database. Turn on the RDS automatic backup function, set the backup in the early morning every day, and keep the backup for 7 days to prevent data loss. [11]

3.2 Experimental Environment Mirror Production

Taking the "Java Programming experiment" as an example, the steps to make the experiment environment mirror are as follows:

(1) Create a temporary ECS instance (specification g6.large.2, image CentOS 7);

(2) Remote login example, install Java JDK 1.8 (configure the environment variables JAVA_HOME), 2023-09 Eclipse, MySQL 8.0 (create the database with the user used);

(3) Configure the experiment environment: preset the Java project template in Eclipse, import the initial data required for the experiment in MySQL, and close unnecessary system services to save resources.

(4) Log in to the enterprise cloud console, enter the "Image service" page, select the temporary ECS instance to create a custom image, name it "Java experimental environment image", and add a description (such as "including JDK 1.8, Eclipse 2023, MySQL 8.0");

(5) Delete the temporary ECS instance, and directly use the custom image when creating the experimental instance in the future. The environment deployment can be completed within 3 minutes.

4. Platform Performance Testing and Verification

Into the way of platform, browser search https://ex7473.huaweicloudsite.cn.

As shown in Figure 1, enter the homepage of the website after entering the URL.

4.1 Testing Environment

In order to ensure the comprehensiveness and representativeness of the test, terminal devices with different brands, different systems and different hardware configurations are selected for this test. The specific configuration information is shown in the table below, covering four core dimensions: terminal type, specific model/system, hardware configuration and browser/tool used for testing.

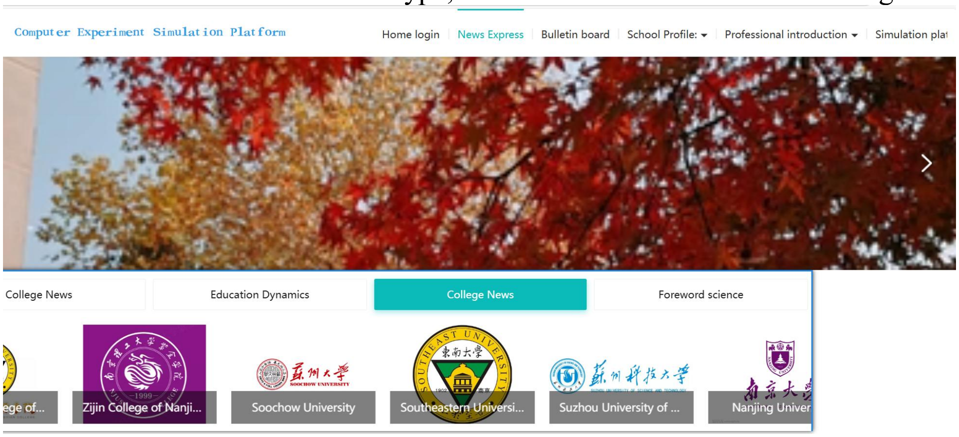


Figure 1. Homepage of the Website

As shown in Figure 2, click Cloud Platform to enter Cloud Host:

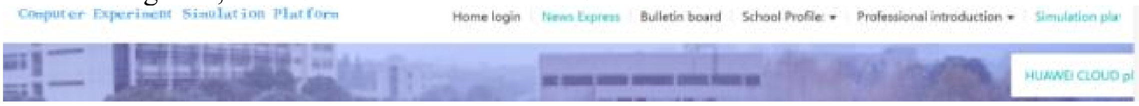


Figure 2. Entry of Cloud Host

4.1.1 Test terminal configuration

Table 1. Terminal Test Table

Termina l types	Specific model/syst em	Hardware configuration	Browser/tools
Computer	Windows 11 25H2	Intel (R) Core (TM) i7-12650H with 32GB of RAM	Microsoft Edge
Computer	Windows 11 25H2	Intel (R) Core (TM) i7-12650H with 32GB of RAM	Microsoft Edge
Mobile phone	Redmi K60	First generation Xiaolong 8+Gen1 octa-core processor with 22G RAM	Xiaomi Browser
Cell phone	Huawei Mate 60, HarmonyOS 4.0	Kirin 9000S with 12GB RAM	Huawei Browser
Tablet	Mi Tablet 7	Xiaomi HyperOS 2 with 8 gigabytes of RAM	Xiaomi Browser
Tablet	Huawei MatePad Air	Gao Tong Xiao Long 888-12 GB RAM	Huawei Browser

4.1.2 Multi-terminal compatibility test results

According to the above terminal configuration, this test sets three core indicators: the success rate of environment startup, the error of code running results, and the fluency of touch screen operation to quantitatively evaluate the compatibility and performance of enterprise cloud platform. The specific test results are as follows.

Table 2. Multi-Terminal Compatibility Test Table

Terminal types	Environment startup success rate (%)	Error in code execution results	Fluency of touch Screen operation
Windows 11	100	0	(Keyboard and Mouse operation)
Windows 11	100	0	(Keyboard and Mouse operation)
Redmi K60 (Generation Xiaolong 8+Gen1)	100	0	Good
Huawei Mate 60 (Hongmeng OS 4.0)	100	0	Good
Mi Tablet 7	100	0	good
Huawei tablet MatePad Air	100	0	Good

4.1.3 Test Conclusion

Combined with the analysis of the test results in Table I and Table II above, the performance of the enterprise cloud platform in the multi-terminal scenario is outstanding, and the core conclusions are as follows:

(1) Comprehensive and reliable compatibility: On different systems such as Windows, HarmonyOS and Xiaomi HyperOS, as well as terminals with different forms of computers, mobile phones and tablets, the success rate of environment startup has reached 100%, and there is no abnormal adaptation problem, showing strong cross-terminal compatibility.

(2) Excellent and stable performance: In all test terminals, the error of the code running results ofHuawei cloud platform is 0, indicating that the platform is accurate and stable in core performance such as data processing and instruction execution, which can meet the needs of various terminals.

(3) Good adaptation of operating experience: for mobile phones, tablets and other touch screen terminals, the fluency of touch screen operation of the platform is evaluated as "good"; Although the computer terminal is mainly operated by keyboard and mouse, and touch screen evaluation is not involved, the overall operation is stable to ensure a high-quality experience under different use scenarios.

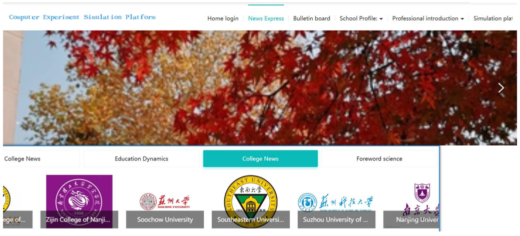


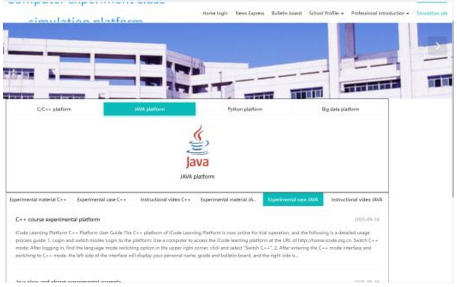
Figure 3. Homepage of the Website

4.2 Example Testing

4.2.1 Enter the experimental environment

Step 1: Enter the website as shown in Figure 3

Step 2: Click simulation to enter the platform in Figure. 4



C++ course experimental platform 2025-09-14

ICode Learning Platform C++ Platform User Guide The C++ platform of ICode Learning Platform is now online for trial operation, and the following is a detailed usage process guide: 1. Login and switch modes Login to the platform: Use a computer to access the ICode learning platform at the URL of http://home.icode.org.cn. Switch C++ mode: After logging in, find the language mode switching option in the upper right corner, click and select "Switch C++". 2. After entering the C++ mode interface and switching to C++ mode, the left side of the interface will display your personal name, grade and bulletin board, and the right side is...

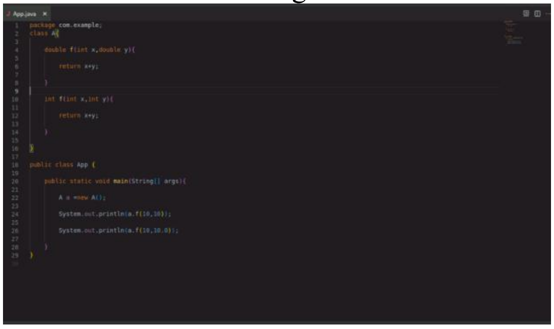
Figure 4. Simulation Platform

4.2.2 Java instance testing

This is shown in Figure 5 below:

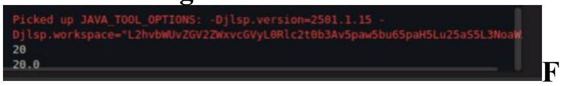
```
package com.example;
class A{
double f(int x,double y){
return x+y;
}
int f(int x,int y){
return x+y;
}
}
public class App {
public static void main(String[] args){
A a =new A();
System. The out. Println (a.f (10, 10));
System. The out. Println (a.f (10,10.0));
}
}
```

The result is shown in Figure VI below:



```
package com.example;
class A{

    double f(int x,double y){

        return x+y;

    }

    int f(int x,int y){

        return x+y;

    }

}

public class App {

    public static void main(String[] args){

        A a =new A();

        System.out.println(a.f(10,10));

        System.out.println(a.f(10,10.0));

    }
}
```

Figure 5. Test Code



```
Picked up JAVA_TOOL_OPTIONS: -Djlsp.version=2501.1.15 -
Djlsp.workspace="L2hvbWUvZGV2ZWxvcGVyL0Rlc2t0b3Av5paw5bu65paH5Lu25aS5L3NoaW
20
20.0
```

F

igure 6. Code Results

5. Platform Advantages

5.1 Platform Advantages

(1) Reduce costs: there is no need to invest in the purchase of local physical servers, but also save the server later hardware maintenance, space occupation and other hidden costs. The platform is based on the enterprise cloud elastic resource architecture, which can flexibly purchase computing, storage and other cloud resources according to the actual needs of experimental course arrangement and the number of students, so as to realize accurate matching and efficient utilization of resources. After practical application verification, compared with the fixed investment model of traditional laboratories, the annual operation and maintenance cost of the platform can be reduced by 40%, which effectively reduces the funding pressure of colleges and universities.

(2) Improve efficiency: According to the scene needs of different majors and different experimental projects, the platform supports custom experimental environment images. Teachers can configure the image file containing the full set of components such as operating system, experimental software and drivers in advance. Students do not need to build the environment step by step, and can complete the rapid deployment of the experimental environment with a key, which greatly shortens the environment preparation time. This feature allows teachers and students to focus more on experimental operation and knowledge exploration, and the overall efficiency of experimental teaching is significantly improved.

(3) Enhanced flexibility: The platform supports remote access to computers, tablets and other terminals. Students can access the experimental environment anytime and anywhere, no matter they are on campus, at home or in other places. This advantage completely breaks through the time and geographical limitations of traditional laboratories, which not only meets the needs of students to arrange their learning progress independently, but also provides a solid support for colleges and universities to carry out mixed teaching models such as online teaching, offline practical operation and remote collaborative experiments, and ADAPTS to diverse teaching scenarios.

(4) Ensure security: Relying on the mature security system of enterprise cloud, the platform constructs a comprehensive and multi -level security protection mechanism. The security group rules were used to accurately control network access rights and effectively resist external network attacks. The combination of regular automatic backup and manual backup was used to ensure that the experimental data was not lost. With the help of IAM authority management function, fine-grained operation permissions were assigned according to different roles such as teachers, students, administrators, etc., to prevent data leakage and misoperation. Multiple security mechanisms work together to provide a reliable guarantee for the integrity and confidentiality of the experimental data and the stable operation of the experimental environment.

5.2 Typical Application Scenarios

(1) Mobile office and fragmented development: developers can quickly write code and debug programs through mobile phones or tablets in scenarios such as commuting and business trips, and synchronize them to the computer for subsequent development.

(2) Cross-terminal computer teaching: university teachers deploy teaching environment and case code on the computer side, and students access to learn and submit homework anytime and anywhere through mobile phones and tablets, realizing "online + offline" hybrid teaching.

(3) Multi-team collaboration in small and medium-sized enterprises: different team members use different terminals (such as computers for developers, tablets for product managers, and mobile phones for testers) to collaborate in real time through the platform to share development progress and code resources.

(4) Rapid contribution to open source projects: open source project contributors do not need to configure a complex environment locally, and can participate in the development through any terminal access platform, reducing the threshold for project participation.

6. Summary

This paper constructed a computer experiment platform and site based on enterprise cloud technology, which took elastic cloud host as the core, integrated OBS, RDS, mirror service and other enterprise cloud services, and realized the rapid deployment of experimental environment, efficient resource management and online teaching process. Through the teaching practice, the platform effectively solved the problems of the traditional experimental teaching, such as insufficient hardware, tedious deployment, and difficult cross-regional collaboration. The experimental completion rate was increased by 15-20 percentage points, and the resource utilization rate was improved.

The future research directions include: first, optimize network adaptation, cooperate with the school network center to improve bandwidth stability, and explore the scheme of offline experimental environment. The second is to reduce the cost of high-order experiments, apply for preferential resources of enterprise cloud education, and expand the application scope of GPU-based ECS instances. The third is to simplify the operation process, develop the experimental environment access client, and reduce the threshold for students to operate. It is believed that with the development of cloud computing technology, the computer experimental platform based on enterprise cloud will be popularized and applied in more colleges and universities, providing strong support for the informatization reform of experimental teaching.

References

[1] Huawei Technologies Co. Huawei cloud elastic cloud server ECS user guide. [2024-03-15]. https://support.huaweicloud.com/ecs/index.html.

[2] Li Na, Zhang Yong. Research on Standardization of Computer Experimental Environment Based on Docker and Cloud Platform. Information Technology, 2022, 46 (7): 145-149. (LI N, ZHANG Y. Research on standardization of computer experimental environment based on Docker and cloud platform. Information Technology, 2022, 46 (7): 145-149.)

[3] Zhang Ming, Li Gang, Wang Yan. Construction and Practice of Computer Experimental Teaching Platform Based on Cloud Computing. Computer Education, 2022, (8): 168-172. (ZHANG M, LI G, WANG Y. Construction and practice of computer experiment teaching platform based on cloud computing. Computer Education, 2022, (8): 168-172.)

[4] Liu Wei, Chen Jing. Research on the construction of distributed experimental environment based on cloud host. Laboratory Research and Exploration, 2021, 40 (11): 265-268. (LIU W, CHEN J. Research on construction of distributed experimental environment based on cloud server. Research and Exploration in Laboratory, 2021, 40 (11): 265-268.)

[5] Wang Jian, Zhao Liang. Design and Implementation of Cloud Platform for Computer Experimental Teaching in Colleges and Universities. Computer Application and Software, 2023, 40 (5): 312-318.

[6] Rao Wenbi, Wang Yunhua, Yang Yanchao, et al. Based on the cloud platform of open experimental teaching and management mode research. Journal of computer education, 2016 (10): 4. DOI: 10.3969 / j.i SSN. 1672-5913.2016.10.040. (Rao Wenbi, Wang Yunhua, Yang Yanchao, et al. Research on the open experimental teaching and management mode of computer science based on cloud platform. Computer Education, 2016(10): 4. DOI: 10.3969/j.issn.1672-5913.2016.10.040.)

[7] Zeng Mingxing, Zhou Qingping, Cai Guomin, et al. Experiments based on the campus of private cloud software engineering practice. Journal of research and practice of integrated platform experiment technology and management, 2015, 32 (4): 5. DOI: 10.3969 / j.i SSN. 1002-4956.2015.04.040.

[8] Wang Feng, Huang Gang. Based on the cloud platform of computer experimental teaching center construction. Journal of experimental technology and management, 2014, 31 (12): 4. DOI: CNKI: SUN: SYJL. 0.2014-12-035.

[9] Zhang Hongli, Shi Jiantao, Zhai Jianhong. Computer Experimental Teaching Platform Based on Cloud Environment. Intelligent Computer and Application, 2015, 005 (3): P. 33-36.

[10] Huawei Cloud. Huawei Cloud Object Storage Service (OBS) Technical White Paper. Shenzhen: Huawei Technologies Co., Ltd., 2023: 12-18.

[11] Huawei Cloud. Huawei Cloud Relational Database Service (RDS) Best Practices. [2024-03-20]. https://support.huaweicloud.com/rds/index.html.