解决方案实践

东软智慧教育云解决方案实践

文档版本 1.0

发布日期 2024-01-25

华为技术有限公司

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

安全声明

漏洞处理流程

华为公司对产品漏洞管理的规定以“漏洞处理流程”为准，该流程的详细内容请参见如下网址：
https://www.huawei.com/cn/psirt/vul-response-process
如企业客户须获取漏洞信息，请参见如下网址：
https://securitybulletin.huawei.com/enterprise/cn/security-advisory

目 录

1 方案概述

应用场景

为满足学校对业务过程处理的高追求和对市场需要的及时响应，出现了一系列以“快速满足需求”为中心的业务系统。然而，这些以“快速满足需求”为中心的业务系统，由于其业务特点的各异性，往往采用不同的技术实现、系统和数据规范，这不仅增加了教育信息化环境的复杂性、业务信息的不对称、不一致性、分散性和无全局性，还增加了运行和维护部门的负担，分散了学校关注支持新业务需求的精力。

因此，从全局规划角度出发的标准化和流程化工作应运而生，并且被提到了教育发展的战略高度。教育部门正不断实施各种信息采集、整理，建立公共数据开放机制，促进教育部门间数据交换，同时向社会开放数据，推动数据的开放和共享；而数据集成作为这些工作的基础设施，将起到支撑的作用。

方案部署架构

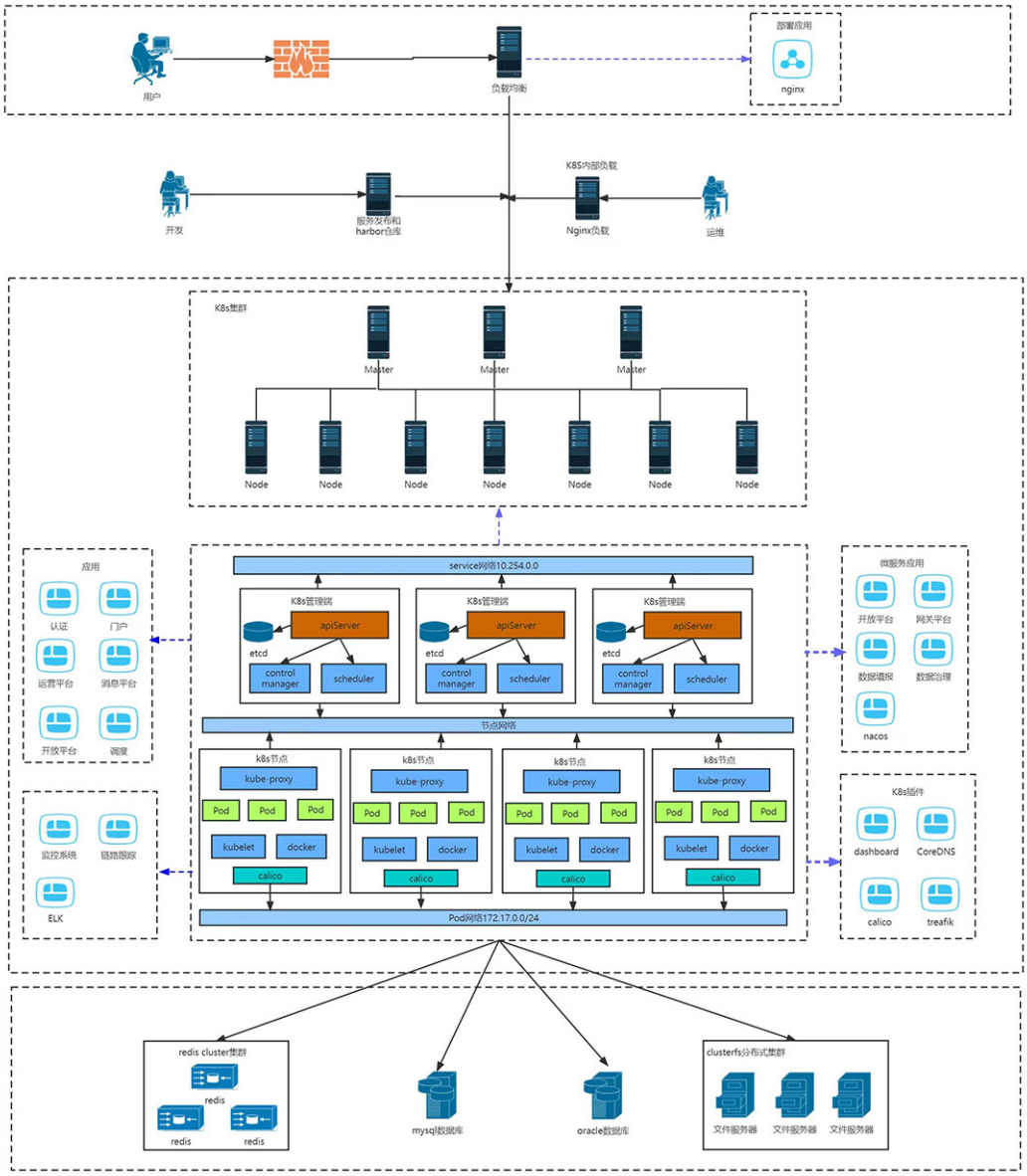


图1-平台部署架构

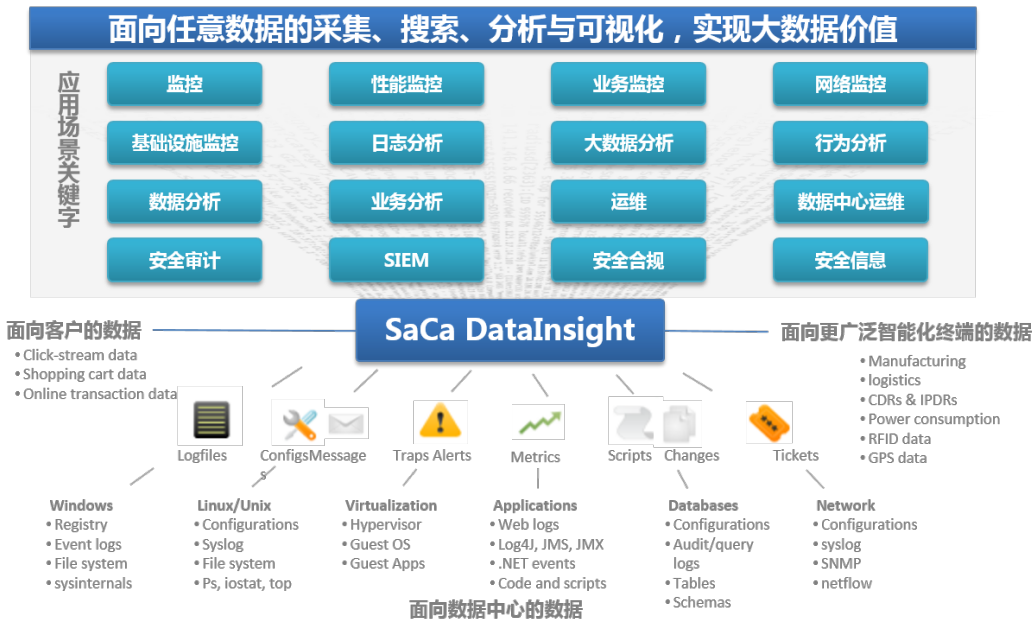
部署架构围绕前后端分离技术，前端采用渐进式前端技术框架vue+webpack+自研组件，要保证模式样式美观，多端统一，后端需要采用微服务架构为支撑，并采用流式消息驱动技术保证数据处理安全、及时、高效。集合大数据存储方案，可以支撑海量数据分布式存储分析，并可通过统一运维平台进行统一监控分析。

平台采用容器化、虚拟化技术支撑，平台的各个业务功能通过微服务化拆分为数十个的支持微服务，每一个微服务均是独立的单元，可以通过微服务的水平扩展形成功能内聚的服务群，满足高并发，高可用的应用要求。微服务架构比传统的应用程序更有效地利用计算资源，当平台的用户数量激增、微服务资源利用率达到瓶颈时，可以通过增加微服务的方式，进一步保障平台性能。当单个微服务出现问题，其他服务仍可继续工作，也可以通过集中微服务资源保障重点业务运行，满足平台对性能和稳定性的苛刻要求。

方案优势

本方案是针对通用数据处理与分析、智慧教育业务应用等推出的智慧教育云平台。它基于先进的大数据处理最佳实践进行构建，能够广泛地适用于各教育领域，如下图所示。

图 1-1 全面发掘数据价值



本方案的关键价值体现如下：

- 统一的数据融合管理平台

  作为一款能够实时采集处理任意数据的平台化工具，它可以帮助学校处理散布在各处的任意协议、内容、格式的异构数据，将分散的数据统一收集并存储管理为后续的数据分析做好基础准备。它提供开箱即用的数据采集和预处理工具，帮助客户可以敏捷地快速搭建分布式数据收集网络，并且确保海量高速数据的处理性能。

- 强大的工具化数据分析平台

  在统一管理海量的各种数据基础之上，对上层提供开箱即用的GUI即席查询工具。在此之上用户使用数据查询DSL就可以直接进行灵活、高度定制化的数据查询工作。即席查询工具还提供了丰富的数据可视化功能以提供更有表现力的数据分析结果。

- 丰富的数据分析应用

  为了适应更广泛的应用领域场景，应对不同层次的应用需求，提供丰富的、功能强大的数据API和应用开发体系规范。使得学校可以面向更深层次、更专业化的领域业务灵活构建更加匹配需求的数据分析应用。

2 资源规划

硬件资源准备

表 2-1 资源准备列表

序号	内存	CPU	磁盘	硬盘空间分区	用途	备注
1	32G	16核	1	200GB	智慧教育数据中台（SaCa DataExchange SaCa DataQuality SaCa DataTransform SaCa DataCompare SaCa DataServices SaCa DataCatalog	按实际负载并发量增加硬件资源
2	32G	16核	2	200GB		
3	64G	16核	1	200GB		
4	32G	16核	2	200GB		
5	32G	16核	2	200GB		
6	32G	16核	2	200GB		
7	32G	16核	2	200GB	智慧教育云平台-基础支撑平台（综合信息门户 统一身份认证 岗位角色平台）	
8	32G	16核	2	200GB		
9	32G	16核	2	200GB		
10	32G	16核	2	200GB		
11	32G	16核	2	200GB	智慧教育云平台-校园管理平台（教学管理服务 校园日常服务 学生日常管理服务 综合素质评价系统）	
12	32G	16核	2	200GB		
13	32G	16核	2	200GB		
14	32G	16核	2	200GB		

软件准备

根据实施方案编制软件安装调试计划。在硬件环境就绪后，按照计划将系统软件进行安装部署与调试，主要内容包括：

- 服务器环境的检查确认；
- 系统软件程序的安装部署与记录；
- 对软件进行配置，并测试参数的正确性，对不合适的配置参数进行调整；
- 现场向用户方工程师进行培训，介绍软件安装调试、使用方法，并给予指导；
- 安装及调试完成后，进行连续测试，确保软件正常运行。

表 2-2 专用软件清单

模块	软件项类型	版本号	软件包名称
统一用户认证管理、数字校园统一信息门户、数据填报组件、教育数据管理工具、教育业务管理门户、智慧校园应用	数据库	V8R6	KINGBASE集群版
	操作系统	V10	银河麒麟
	中间件	7.0V	东方通
	管理服务	10.3.0	平台管理端
	WEB服务	10.3.0	平台WEB端
	备份服务	10.3.0	平台备份端
	动态服务	2.0.0	应用开放平台

表 2-3 通用与支撑软件清单

软件项名称	软件项描述	
	用途	版本
操作系统	客户端	Windows系统
浏览器	客户端	360浏览器

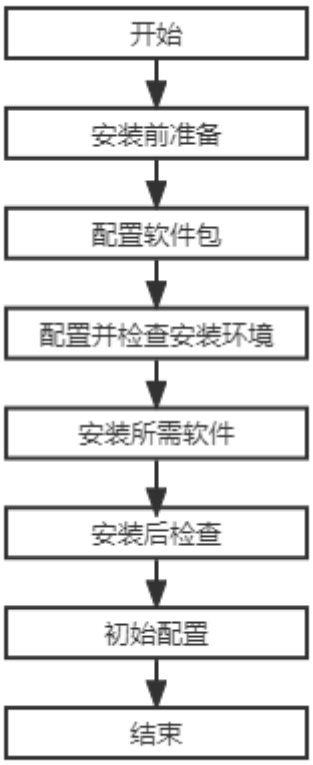
3 方案实施计划和步骤

3.1 实施流程及计划

表 3-1 实施计划表

序号	任务	负责人	工期（天）
1	基础环境及k8s搭建	xxx	15
2	应用环境部署	xxx	15

图 3-1 软件安装流程图



3.2 实施步骤

3.2.1 安装前环境检查

参考安装部署及组网规划，检查机房环境情况以及网络配置、硬件配置情况，对软件安装前进行环境检查。

表 3-2 环境检查列表

序号	内容	是否满足	备注
1	操作系统版本	是	/
2	cpu	是	/
3	内存	是	/
4	交换空间及硬盘等资源配置	是	/
5	服务器网络	是	/
6	能否通外网（安装数据库需要）	是	/

3.2.2 软件安装

智慧教育数据中台、基础支撑平台（统一用户认证管理、数字校园统一信息门户、岗位角色管理）、校园管理平台。

安装前环境配置

操作系统初始化工作。

```
## 配置系统
###### 关闭 防火墙
systemctl stop firewalld
systemctl disable firewalld
###### 关闭 SeLinux
setenforce 0
sed -i "s/SELINUX=enforcing/SELINUX=disabled/g" /etc/selinux/config
###### 关闭 swap
swapoff -a
yes | cp /etc/fstab /etc/fstab_bak
cat /etc/fstab_bak |grep -v swap > /etc/fstab
###### yum epel源
yum install wget telnet -y
mv /etc/yum.repos.d/CentOS-Base.repo /etc/yum.repos.d/CentOS-Base.repo.backup
wget -O /etc/yum.repos.d/CentOS-Base.repo “镜像源地址”/Centos-7.repo
wget -O /etc/yum.repos.d/epel.repo “镜像源地址”/epel-7.repo
yum clean all
yum makecache
###### 修改 /etc/sysctl.conf
modprobe br_netfilter
cat <<EOF >  /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1
net.ipv4.ip_forward = 1
vm.swappiness=0
EOF
sysctl -p /etc/sysctl.d/k8s.conf
###### 开启 ipvs
cat > /etc/sysconfig/modules/ipvs.modules <<EOF
#!/bin/bash
modprobe -- ip_vs
modprobe -- ip_vs_rr
modprobe -- ip_vs_wrr
modprobe -- ip_vs_sh
modprobe -- nf_conntrack_ipv4
EOF
chmod 755 /etc/sysconfig/modules/ipvs.modules && bash /etc/sysconfig/modules/ipvs.modules && lsmod |
grep -e ip_vs -e nf_conntrack_ipv4
# 设置 yum repository
yum install -y yum-utils \
device-mapper-persistent-data \
lvm2
yum-config-manager --add-repo “镜像源地址”/docker-ce.repo
# 安装并启动 docker
yum install -y docker-ce-18.09.8 docker-ce-cli-18.09.8 containerd.io
# 添加ipvs支持
yum install -y nfs-utils ipset ipvsadm
```

etcd 集群软件安装

表 3-3 etcd 集群软件安装

在master1上安装cfssl	wget https://pkg.cfssl.org/R1.2/cfssl_linux-amd64 wget https://pkg.cfssl.org/R1.2/cfssljson_linux-amd64 chmod +x cfssl_linux-amd64 cfssljson_linux-amd64 mv cfssl_linux-amd64 /usr/local/bin/cfssl mv cfssljson_linux-amd64 /usr/local/bin/cfssljson
安装etcd二进制文件	# 创建目录 mkdir -p /data/etcd/bin # 下载 cd /tmp wget https://storage.googleapis.com/etcd/v3.3.25/etcd-v3.3.25-linux-amd64.tar.gz tar zxf etcd-v3.3.25-linux-amd64.tar.gz cd etcd-v3.3.25-linux-amd64 mv etcd etcdctl /data/etcd/bin/
创建ca证书，客户端，服务端，节点之间的证书	Etcd属于server ,etcdctl 属于client，二者之间通过http协议进行通信。 创建目录 创建ca证书 生成客户端证书 cfssl gencert -ca=ca.pem -ca-key=ca-key.pem -config=ca-config.json -profile=client client.json | cfssljson -bare client - 生成server，peer证书 cfssl gencert -ca=ca.pem -ca-key=ca-key.pem -config=ca-config.json -profile=server etcd.json | cfssljson -bare server cfssl gencert -ca=ca.pem -ca-key=ca-key.pem -config=ca-config.json -profile=peer etcd.json | cfssljson -bare peer 将master01的/data/etcd/ssl目录同步到master02和master03上 scp -r /data/etcd etcd2:/data/etcd scp -r /data/etcd etcd3:/data/etcd
etcd systemd配置文件	vim /usr/lib/systemd/system/etcd.service 三台主机配置不一样用的时候把注释尽量删除
启动etcd集群	systemctl daemon-reload systemctl enable etcd systemctl start etcd systemctl status etcd

安装 kubeadm，kubelet，kubectl

所有节点安装kubeadm，kubelet。kubectl是可选的，可以安装在所有机器上，也可以只安装在一台master1上。

```
yum install -y kubelet kubeadm kubectl
```

在所有安装kubelet的节点上，将kubelet设置为开机启动

```
systemctl enable kubelet
```

初始化 master

表 3-4 初始化 master

在master1上将搭建etcd时生成的的ca证书和客户端证书复制到指定地点并重命名	mkdir -p /etc/kubernetes/pki/etcd/ #etcd集群的ca证书 cp /data/etcd/ssl/ca.pem /etc/kubernetes/pki/etcd/ #etcd集群的client证书，apiserver访问etcd使用 cp /data/etcd/ssl/client.pem /etc/kubernetes/pki/apiserver-etcd-client.pem #etcd集群的client私钥 cp /data/etcd/ssl/client-key.pem /etc/kubernetes/pki/apiserver-etcd-client-key.pem
创建初始化配置文件	kubeadm config print init-defaults > kubeadm-init.yaml
执行初始化	kubeadm init --config=kubeadm-init.yaml
配置kubectl	要使用 kubectl来 管理集群操作集群，需要做如下配置 mkdir -p $HOME/.kube sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config sudo chown $(id -u):$(id -g) $HOME/.kube/config 测试下，kubectl是否正常，需要注意此时master1的notready状态是正常的，因为尚未部署网络插件 [root@master01] # kubectl get node NAME STATUS ROLES AGE VERSION master01 NotReady master 66s v1.19.2
添加master02和master03	首先将 master1中的生成的集群共用的ca证书，scp到其他master机器 scp -r /etc/kubernetes/pki/* master02:/etc/kubernetes/pki/ 将初始化配置文件复制到master2 scp kubeadm-init.yaml master02:/root/ 初始化master2 修改后初始化具体修改内容根据上面的标准文件注释修改 Kubeadm init --config=kubeadm-init.yaml

将 worker 节点加入集群

在master01上生成加入key

```
kubeadm token create --print-join-command
```

在其他worker节点执行

```
ubeadm join master:6443 --token abcdef.0123456789abcdef \
--discovery-token-ca-cert-hash \
sha256:fb4e252253b55974edff65cb4765e9979f8785cd67a6ed41f87c83c6bcc3ac4a
```

将代码打包到 git 仓库

图 3-2 编译好的工程文件通过 docker build 进行编译

```
[root@harbor cloud]# docker build . -t harbor.neusoftdgjy.com/app/cloud:v1.00
Sending build context to Docker daemon  206.1MB
Step 1/8 : FROM harbor.od.com/public/jre:8u112
Get "https://harbor.od.com/v2/": dial tcp: lookup harbor.od.com on 10.152.169.2:53: no such host
[root@harbor cloud]# docker build . -t harbor.neusoftdgjy.com/app/cloud:v1.0
Sending build context to Docker daemon  206.1MB
Step 1/8 : FROM harbor.neusoftdgjy.com/app/jre8:v1.0
 ---> fa3a085d6ef1
Step 2/8 : RUN /bin/cp /usr/share/zoneinfo/Asia/Shanghai /etc/localtime &&    echo 'Asia/Shanghai' >/etc/timezone
 ---> Using cache
 ---> d8b2237caea0
Step 3/8 : ADD models /opt/models
 ---> c67ee3f4fd71
Step 4/8 : COPY xjar-agent-hibernate-v1.0.0.jar /opt/
 ---> 65741107c526
Step 5/8 : COPY cloud-sys-x.jar  /opt/
 ---> 465d36d1f507
Step 6/8 : WORKDIR /opt
 ---> Running in db49cbc0e895
Removing intermediate container db49cbc0e895
 ---> ddb74c1a4522
Step 7/8 : ADD entrypoint.sh /opt/entrypoint.sh
 ---> ab08e0630919
Step 8/8 : CMD ["/opt/entrypoint.sh"]
 ---> Running in 60fb6347b79b
Removing intermediate container 60fb6347b79b
 ---> ddbb9e304322
Successfully built ddbb9e304322
Successfully tagged harbor.neusoftdgjy.com/app/cloud:v1.0
```

图 3-3 通过 docker push 将工程文件打包到 git 仓库

```
[root@harbor cloud]# docker push harbor.neusoftdgjy.com/app/cloud:v1.0
The push refers to repository [harbor.neusoftdgjy.com/app/cloud]
5ce4c19b0d93: Pushed
2118530fd530: Pushed
1f9d44620de8: Pushed
d53db2a92aed: Pushed
a6f63d5afe1b: Mounted from sems/sample
0690f10a63a5: Mounted from app/jre8
c843b2cf4e12: Mounted from app/jre8
fddd8887b725: Mounted from app/jre8
42052a19230c: Mounted from app/jre8
8d4d1ab5ff74: Mounted from app/jre8
v1.0: digest: sha256:c3f8c585b374a8f36197f938b48e003e729f89935c646b34ff4bb21d2d8d1e2e size: 2417
```

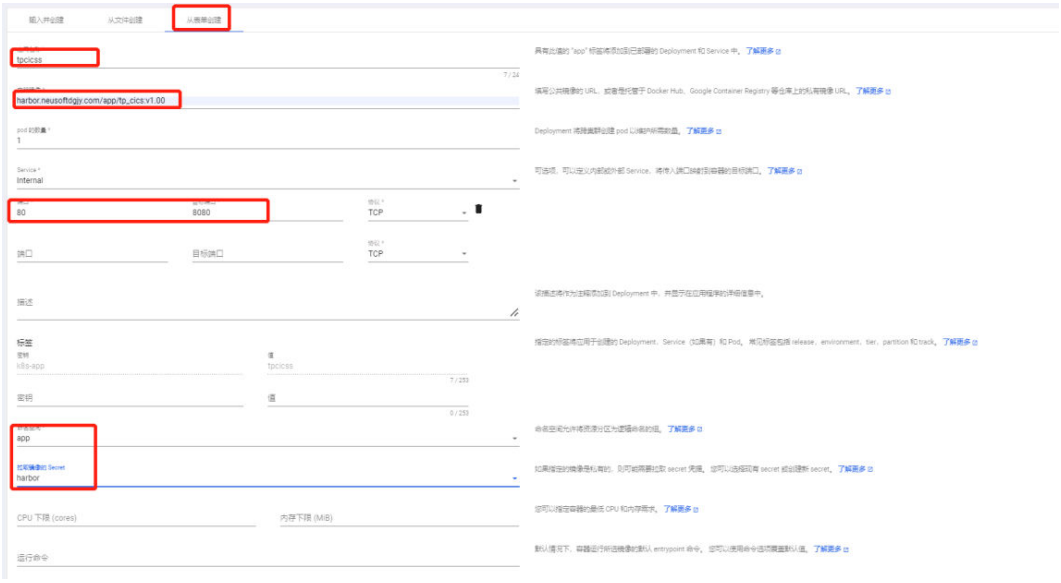
K8s 应用部署

K8s登录后，单击左侧“Deployments”菜单，单击右侧的“+”按钮，可进入到应用添加页面。

图 3-4 进入应用添加页面

进入到应用添加页面，填入应用名，输入git仓库名称、工作区、端口映射，单击保存，完成应用添加，如下图：

图 3-5 应用添加页面



3.2.3 数据库导入

智慧教育数据中台、基础支撑平台（统一用户认证管理、数字校园统一信息门户、岗位角色管理、）、校园管理平台。

表 3-5 数据库导入步骤

数据库导入步骤	命令
进入数据库创建目录、用户表空间和用户	sqlplus / as sysdba(下面依次输入以下命令后回车) > create directory gjtdb as '/home/oracle/'; > create tablespace tp datafile '/oracle/oradata/ORCL/orclpdb1/tp01.dbf' size 50m autoextend on next 50m maxsize unlimited > create user tp_sems identified by ****** default tablespace tp temporary tablespace temp; > grant dba to tp_sems; > exit
上传准备好的数据备份dmp文件，用导入命令进行导入	impdp tp_sems/******** directory=gjtdb dumpfile= gjtdb.dmp full=y
等待命令执行完成后数据就成功导入	/

3.2.4 应用联调

在软件系统分别安装调试完成之后，需要进行充分的系统测试、联调测试及配置优化才能提供给用户方使用。因此，需要对系统的各个组成部分进行充分的测试，并按照实际测试结果调整、优化相关设备或软件的配置。具体内容有：

- 根据与用户方的沟通，制定系统联调及调优计划，同时请用户方做好内部相关部门的协调工作，配合实施系统联调和优化。

- 联调测试工作完成后，形成系统联调及调优报告，作为重要项目管理文档记录。

系统联调优化工作可能会涉及到多部门配合，需要用户方能提前做好协调好工作，协助系统联调优化工作的顺利实施，如：

- 服务组件功能调测：服务组件基本功能调测。
- 服务组件到接入数据的联调：从第三方系统或者底层数据仓库加载原始数据，汇聚、分析后写入到服务组件依赖的数据库。
- 第三方应用与服务组件对接：调测应用是否可以通过接口从服务组件获取数据。
- 第三方应用调测：第三方应用场景调测，页面及API在具体应用场景下的调测。

4 软件系统健康自检

系统状态检查

- **系统运行状态检查**

  K8s登录后，单击pod，教育云k8s同时运行18个pod，发布后需要对新发布的pod进行检查。如果是绿色，则视为健康；如果不是绿色，则视为不健康，需要对部署的应用进行检查。

  表 4-1 系统运行状态检查

检查内容	是否正常	备注
服务器cpu	是	/
服务器内存	是	/
网络资源	是	/
数据库运行状态	是	/
nginx运行状态	是	/

- **系统告警信息检查**

  检查系统告警情况，无异常告警。

- **系统基本功能检查**

  登录系统，检查各模块基本功能是否正常。

  表 4-2 各模块功能检查列表

模块	功能模块	是否正常	备注
智慧教育数据中台	SaCa DataExchange	是	/
	SaCa DataQuality	是	/
	SaCa DataTransform	是	/
	SaCa DataCompare	是	/

模块	功能模块	是否正常	备注
	SaCa DataServices	是	/
	SaCa DataCatalog	是	/
智慧教育云平台-基础支撑平台	综合信息门户	是	/
	统一身份认证	是	/
	岗位角色平台	是	/
智慧教育云平台-校园管理平台	教学管理服务	是	/
	校园日常服务	是	/
	学生日常管理服务	是	/
	综合素质评价系统	是	/

系统信息收集及自检报告

表 4-3 系统检查清单

名称	检查项目	检查记录	改善建议
Nginx	进程是否启动	正常	无
	监听端口是否启动	正常	无
	日志是否有正常访问记录	正常	无
	页面访问是否正常	正常	无
应用程序	能否正常登录	正常	无
	登录后是否按照角色显示不同权限菜单	正常	无
	能否正常录入数据	正常	无
	管理员能否正常查看信息	正常	无
日志监控	进程是否启动	正常	无
	监听端口是否启动	正常	无
	是否可以正常打开监控页面，查看日志监控情况	正常	无

备份与恢复策略

关键数据、关键配置文件、关键日志必须有例行的备份与快速恢复机制。

在文件或数据丢失、服务器损坏、存储损坏等场景下，要能快速找到有效的备份文件，快速恢复，数据偏差不超过24小时。

5 修订记录

发布日期	修订记录
2024-1-25	内容修订
2023-11-15	第一次正式发布。