EMR工作内容交接

## 工作交接安排

|  |  |  |
| --- | --- | --- |
| 模块 | 工作内容 | 交接人 |
| Java代码 | Hive、HBase资源操作相关BSS以及监控 | 辛杰煊 |
| HUE相关 | HUE源码以及相关修改 | 辛杰煊 |
| 镜像相关 | Emr服务部署所需镜像及数据 | 辛杰煊 |
| Ambari集成EMR相关 | Ambari集成EMR服务代码及部署 | 辛杰煊 |

交接人同为项目开发者，全程参与EMR服务的开发、测试与部署，对emr项目有深入的理解。

注意事项：

EMR服务有多种部署方案，从前期的政务云部署、到独立部署、再到与托管Hadoop融合集成到Ambari中的独立部署和云平台部署。以上这些部署模式中，使用的emr-api和emr-service模块代码基本相同，部署环境、部署方式以及部署工具会有差别。

## 项目组成

Emr-api模块:

Git地址： http://git.inspur.com/Insight-HD/dataspace/userspaceapi

功能：

EMR服务由于部署方式的原因，采用模块化设计，api模块主要承担前端请求的处理、业务数据的存储以及service模块的通信。

代码模块：

Hive、HBase的业务逻辑、BSS对接以及监控这几个方面由我负责开发。其中，Hive与HBase业务逻辑与HDFS以及YARN的业务处理逻辑大致相同，都是api接收前端请求存储业务数据并将资源操作请求转发至service模块。

BSS监控的业务逻辑主要是由类BssOrderListener处理，会使用到JsonMessGenerator这个工具类。这部分的详细说明文档在confluence上<http://10.10.7.5:8090/pages/viewpage.action?pageId=22396949>

云监控由com.inspur.cloud.dataspace.metrics下的代码处理业务逻辑，MonitorMetricServiceImpl实现类中调用com.inspur.cloud.monitor包下的CloudMonitorAccessor类（需要注入此类）。云监控功能依赖集群安装Ambari Metrics模块，否则云监控无法取到集群组件的状态信息。Ambari监控Hadoop集群jar包使用相关说明http://10.10.7.5:8090/pages/viewpage.action?pageId=24180621，git地址<http://git.inspur.com/Insight-HD/insight-hd/ambari-component-monitor->。监控demo <http://git.inspur.com/LMA/prometheus-java-sdk-demo> 。

Emr-service模块:

Git地址： http://git.inspur.com/Insight-HD/dataspace/userspaceservice

功能：

service模块用于接受api模块的请求连接Hadoop集群与集群进行交互，操作集群上的HDFS、YARN、Hive、HBase资源以及Ranger上的权限等。

代码模块：

Hive以及HBase业务扩及由我开发完成，包括新增、删除和改配。其中配额控制包括空间配额与表数量配额。HBase的空间配额和表配额均是调用org.apache.hadoop.hbase.client包中的Admin接口实现，同时需在集群中hbase-site中添加hbase.quota.enabled和hbase.quota.refresh.period两个属性（“浪潮云EMR部署手册”中有详细说明）。Hive的空间配额管控是借助hdfs接口实现（需将hive加入hdfs的超级用户组中，“浪潮云EMR部署手册”中有详细说明），空间用量的查询借助org.apache.hadoop.fs包中getUsed()方法实现，使用时需要缓存hdfs票据信息。Hive表数量配额管控是借助额外jar包实现，同时需要修改集群中hive-site中相关配置（“浪潮云EMR部署手册”中有详细说明），hive-quota-plugin的jar包git地址<http://git.inspur.com/Insight-HD/dataspace/hive-quota-plugin>。

Hue模块：

Git地址：<http://git.inspur.com/Insight-HD/dataspace/emr-hue>

功能：

HUE是Hadoop集群任务管理的可视化平台。目前，emr-hue项目已经集成了hdfs，hive，hbase以及oozie。HUE是Django框架编写，想要进行代码调试需安装python及IDE，这里推荐Pycharm。HUE代码一般需要安装到远端集群上，可以选择远程调试。在Pycharm/File/Settings/Build,Execution,Deoloyment/Deployment下添加sftp服务，将HUE代码安装节点host,username/password,rootpath等信息填写完毕，测试正常连接，就能实现本地和远端代码的同步上传和下载。VNC是一款linux桌面客户端，在HUE代码安装节点安装Pycharm与VNC实现HUE代码的调试。

10.221.129.13与10.221.129.14上均已安装Pycharm，VNC以及HUE代码，能够实现HUE代码的调试。以13节点为例，使用步骤：1.通过VNC连接到远程桌面（root/bigdata123?）;2.在Application中打开terminal命令行窗口，进入/root/pycharm/pycharm-2018.1.4/bin目录运行pycharm.sh等待PycharmIDE打开；3.进入IDE后点击File/Open输入/root/hue/emr-hue/打开已经编译好的HUE代码；4.点击Run运行代码。注意编译HUE代码参考<https://docs.gethue.com/developer/development/>；

包括HUE与Keycloak集成<http://10.10.7.5:8090/pages/viewpage.action?pageId=7777205> ；常见问题<http://10.10.7.5:8090/pages/viewpage.action?pageId=22412866>。

HUE的核心配置文件位于emr-hue/desktop/conf/pseudo-distributed.ini文件，包括Webserver监听的host与port，包括Debug模式，包括应用黑名单（HUE客户端不显示的应用）以及安装hdfs,yarn,hive,hbase,zookeeper,oozie等组件的配置，详细参考ambari-server\src\main\resources\stacks\HDP\3.1\services\EMR\package\templates\hue.pseudo.distributed.ini.j2文件（ambari集成emr服务代码）。

镜像相关：

镜像仓库：<http://10.221.129.22/InspurHD1.0/hdp/emr/>

以及本地源117.73.10.103:/var/www/html/emr（root/ Bigdata123）

功能：

制作这些镜像是为了实现emr服务的独立部署。包括Nginx,Mysql,emr-api,emr-sercive,emr-hue等相关8个镜像，使用docker-compose作为镜像管理工具。docker-compose设置为开机自启，保证服务的可用性。mysql中的数据实现定时备份在宿主机，实现用户数据的安全。这些在EMR部署手册中均有详细介绍。

关于docker安装与使用以及常见问题处理，可以参考<http://10.10.7.5:8090/pages/viewpage.action?pageId=24177873>

关于Nginx反向代理配置文件参数设置以及日志设置，可以参考<http://10.10.7.5:8090/pages/viewpage.action?pageId=24175486>

关于EMR服务部署镜像的说明，云平台部署与独立部署的详细步骤，可以参考EMR部署手册<http://10.10.7.5:8090/pages/viewpage.action?pageId=22405806>

关于Hadoop集群部署前环境检查与准备，可以参考<http://10.10.7.5:8090/pages/viewpage.action?pageId=24175478>

关于Hadoop集群的Benchmark性能测试流程，可以参考<http://10.10.7.5:8090/pages/viewpage.action?pageId=24175468>

关于在HUE中添加oozie作业需要参考:/服务安装目录/emr-docker-data/emr-hue/examples/apps/中样例，首先选择作业类型（以map-reduce为例），目录下有如下文件：

|  |
| --- |
| [root@worker map-reduce]# ll  total 20  -rw-r--r-- 1 yarn-ats users 1012 Dec 6 2018 job.properties  -rw-r--r-- 1 yarn-ats users 1028 Dec 6 2018 job-with-config-class.properties  drwxr-xr-x 2 root root 4096 Jan 19 15:22 lib  -rw-r--r-- 1 yarn-ats users 2274 Dec 6 2018 workflow-with-config-class.xml  -rw-r--r-- 1 yarn-ats users 2559 Dec 6 2018 workflow.xml |

重点参考workflow-with-config-class.xml文件，该文件有此项作业配置的详细信息，包括：队列名、作业目录、输入文件路径以及使用的java类，注意delete path，作用是指定每次作业执行前上次作业完成后生成的文件，不删除此文件mr作业无法完成。

lib目录中存放该示例所需使用的jar包，需要放到当前用户的用户目录下。

Ambari集成EMR相关：

Git地址：git.inspur.com/Insight-HD/insight-hd/inspurhd1.0-ambari-2.7.3

功能：

Ambari集成EMR，使得EMR成为Ambari的一个自定义组件，能够实现EMR服务一键式安装部署，简化EMR服务的部署流程。进而EMR可以成为托管Hadoop服务提供多租户管理和资源管控的组件。目前，117.73.8.13集群的Ambari中集成了云平台部署模式的EMR服务，10.221.129.14集群的Ambari中集成了独立部署模式的EMR服务。

代码模块：

EMR服务的代码位于项目的\ambari-server\src\main\resources\stacks\HDP\3.1\services\EMR目录下，整个项目的结构如下：

|  |
| --- |
| [root@manager services]# tree EMR/  EMR/  ├── alerts.json  ├── configuration  ├── kerberos.json  ├── metainfo.xml  ├── package  │   ├── archive.zip  │   ├── scripts  │   └── templates  └── quicklinks  └── quicklinks.json |

configuration目录下存放EMR服务的配置参数，其中emr-api-config.xml是emr-api模块的配置参数；emr-env.xml是EMR服务的配置参数；emr-hue-config.xml是emr-hue模块的配置参数；emr-service-config.xml是emr-service模块的配置参数。

|  |
| --- |
| [root@manager EMR]# tree configuration/  configuration/  ├── emr-api-config.xml  ├── emr-env.xml  ├── emr-hue-config.xml  ├── emr-log4j.xml  └── emr-service-config.xml |

kerberos.json文件是在Kerberos开启后，安装EMR服务时生成hue.keytab所需的配置文件，需要配置的信息有keytab名称与principle以及存放的路径。

metainfo.xml是存放EMR服务的元数据信息的配置文件，包括服务名称、组件名称、运行脚本、安装时长、服务状态检测脚本、配置文件、依赖的组件以及快速连接等信息。

package目录下包含两部分内容：scripts目录下是EMR服务安装及运行所需的python脚本，emr\_service.py负责服务安装时资源下载与安装、服务的初始化、服务的启停等，params.py负责定义服务中的各项参数，repoin.py获取集群repo地址的脚本，service\_check.py负责服务状态检查的脚本，status\_params.py定义服务状态相关参数的脚本。templates目录下是所需替换的服务配置文件的模板，模板文件中的有关参数被{{params.py中定义的变量}}所替代，通过Ambari UI修改配置参数就可以替换到服务中所对应的配置文件中去。

|  |
| --- |
| [root@manager EMR]# tree package/  package/  ├── archive.zip  ├── scripts  │   ├── emr\_service.py  │   ├── emr\_service.py.bak  │   ├── params.py  │   ├── repoin.py  │   ├── service\_check.py  │   └── status\_params.py  └── templates  ├── api.application.prod.yaml.j2  ├── api.application.yaml.j2  ├── docker-compose.yml.j2  ├── hue.pseudo.distributed.ini.j2  ├── iam.apiserver.application.properties.j2  ├── iam.auth.ladon.dbConfig.j2  ├── iam.web.application.json.j2  ├── nginx.default.conf.j2  ├── service.application.dev.yaml.j2  └── web.application.json.j2 |

quicklinks目录下存放有quicklinks.json文件，该文件主要用于配置Ambari UI中快速链接中超链接的相关信息。

|  |
| --- |
| [root@manager EMR]# tree quicklinks/  quicklinks/  └── quicklinks.json |

其他信息可以参考<https://www.jianshu.com/p/96271bccb6bc> 以及<https://cloud.tencent.com/developer/article/1372228>