manager改造接口文档

[manager改造接口文档 1](#_Toc479597013)

[接口约定： 4](#_Toc479597014)

[Project类 4](#_Toc479597015)

[1、add\_project 4](#_Toc479597016)

[2、query\_project 4](#_Toc479597017)

[3、delete\_project 5](#_Toc479597018)

[Graphic类 5](#_Toc479597019)

[1、add\_graphic 5](#_Toc479597020)

[2、require\_graphic 6](#_Toc479597021)

[Database类 6](#_Toc479597022)

[1、save\_db\_info 6](#_Toc479597023)

[2、query\_db\_info 7](#_Toc479597024)

[3、query\_fav\_db 7](#_Toc479597025)

[4、delete\_fav\_db 7](#_Toc479597026)

[5、add\_fav\_db 8](#_Toc479597027)

[6、test\_db\_connection 8](#_Toc479597028)

[7、check\_sourcedb\_env 8](#_Toc479597029)

[Capture组件 9](#_Toc479597030)

[1、save\_parameter 9](#_Toc479597031)

[2、add\_capture\_property 10](#_Toc479597032)

[3、query\_capture\_config 11](#_Toc479597033)

[Queue组件 12](#_Toc479597034)

[1、save\_queue\_info 12](#_Toc479597035)

[2、query\_queue\_info 12](#_Toc479597036)

[3、delete\_queue\_pkg\_statis 12](#_Toc479597037)

[4、query\_queue\_statis 13](#_Toc479597038)

[5、query\_queue\_pkg 14](#_Toc479597039)

[6、delete\_queue\_pkg\_xid 14](#_Toc479597040)

[etl组件 15](#_Toc479597041)

[1、query\_etl\_config 15](#_Toc479597042)

[2、query\_etl\_user 15](#_Toc479597043)

[3、query\_etl\_table 16](#_Toc479597044)

[4、query\_etl\_table\_rule 16](#_Toc479597045)

[5、save\_etl\_table\_rule 16](#_Toc479597046)

[6、query\_column\_list 17](#_Toc479597047)

[7、legal\_check\_expression 17](#_Toc479597048)

[8、query\_etl\_add\_config 17](#_Toc479597049)

[9、save\_etl\_add\_config 18](#_Toc479597050)

[10、query\_etl\_delete\_config 18](#_Toc479597051)

[11、save\_etl\_delete\_config 18](#_Toc479597052)

[12、query\_etl\_map\_config 19](#_Toc479597053)

[13、save\_etl\_map\_config 19](#_Toc479597054)

[14、query\_etl\_condition\_config 19](#_Toc479597055)

[15、save\_etl\_ condition\_config 20](#_Toc479597056)

[16、query\_etl\_transform\_config 20](#_Toc479597057)

[17、save\_etl\_transform\_config 20](#_Toc479597058)

[18、query\_etl\_convert\_config 21](#_Toc479597059)

[19、save\_etl\_convert\_config 21](#_Toc479597060)

[20、query\_etl\_table\_filter 21](#_Toc479597061)

[21、save\_etl\_table\_filter 21](#_Toc479597062)

[22、query\_etl\_operation\_filter 22](#_Toc479597063)

[23、save\_etl\_operation\_filter 22](#_Toc479597064)

[24、query\_batch 22](#_Toc479597065)

[25、save\_batch\_rule 23](#_Toc479597066)

[26、query\_batch\_rule 23](#_Toc479597067)

[27、query\_batch\_checked 23](#_Toc479597068)

[28、save\_batch\_codevalue 24](#_Toc479597069)

[29、query\_etl\_rule\_ptable 24](#_Toc479597070)

[Loader组件 24](#_Toc479597071)

[1、query\_apply\_config 24](#_Toc479597072)

[2、add\_apply\_config 25](#_Toc479597073)

[3、query\_apply\_sourcedb\_object 26](#_Toc479597074)

[TClient/TServer传输组件 26](#_Toc479597075)

[1、query\_qrecv\_config 26](#_Toc479597076)

[2、add\_apply\_config 26](#_Toc479597077)

[3、query\_tserver\_config 27](#_Toc479597078)

[4、add\_tserver\_config 27](#_Toc479597079)

[删除组件delete\_component 27](#_Toc479597080)

[Group类 28](#_Toc479597081)

[1. add\_group 28](#_Toc479597082)

[2. del\_group 28](#_Toc479597083)

[3. modify\_group 28](#_Toc479597084)

[4. reset\_group 29](#_Toc479597085)

[5. query\_group 29](#_Toc479597086)

[System\_setting类 30](#_Toc479597087)

[1. save\_reporter\_conf 30](#_Toc479597088)

[2. save\_monitor\_conf 30](#_Toc479597089)

[3. query\_reporter\_conf 31](#_Toc479597090)

[4. query\_monitor\_conf 31](#_Toc479597091)

[5. dip\_manuser\_query\_user 31](#_Toc479597092)

[6. dip\_manuser\_save\_user 32](#_Toc479597093)

[7. dip\_manuser\_save\_user 32](#_Toc479597094)

[8. dip\_manuser\_save\_user 33](#_Toc479597095)

[Tool\_set类 33](#_Toc479597096)

[1. export\_config 33](#_Toc479597097)

[2. import\_config 33](#_Toc479597098)

[3. export\_errinfo 34](#_Toc479597099)

[Error类 34](#_Toc479597100)

[1. query\_apply\_full\_error 34](#_Toc479597101)

[2. query\_apply\_current\_error 35](#_Toc479597102)

[3. trunc\_apply\_error\_record 36](#_Toc479597103)

[4. marked\_apply\_error 36](#_Toc479597104)

[5. download\_error\_file 37](#_Toc479597105)

## 接口约定：

//全部使用utf-8编码

//JSON请求接口

{

md5:"",   //request内容部分的md5加密值，request内容的字符串的加密结果

request:{} //两种类型，对象{},数组[]

}

//JSON响应接口

{

   status:, //true 成功 false 失败

   md5:"",   //response内容的md5加密值，response内容的字符串的加密结果

   response:{error\_code:"code", error\_msg:"msg"} //三种类型，对象{},数组[],字符串""

}

//前后台分别校验MD5，检查报文完整性。

## Project类

### 1、add\_project

接口名称：新建project

前端发送：

{"name":"name9527","create\_time":"2016/12/12","desc":"desc"}

返回类型：{对象}

{"status":true,"md5":"a5fb1860fb12259807b5e3df779660a6","response":{"project\_id":"project\_78"}}

接口描述：

首先查询数据库表t\_project\_info是否存在名字为name的project，如果存在返回“已存在”的错误，不存在，再检查目前数据库表中已有project的数量是否超出配置文件中配置的最大个数，超出返回错误，否则插入project信息,然后把刚插入的project\_id返回前端。

### 2、query\_project

接口名称：查询project

前端发送：{md5: "99914b932bd37a50b983c5e7c90ae93b", request: {}}

返回类型：【数组】

{"status":true,"md5":"83d9d3ffa84ff381bb6e05adb1ba137e","response":[{"id":"project\_2","name":"test","create\_time":"2017-01-05","desc":"test"},{"id":"project\_3","name":"test1","create\_time":"2017-01-05","desc":"test1"},{"id":"project\_4","name":"test2","create\_time":"2017-01-05","desc":"test2"},{"id":"project\_5","name":"test3","create\_time":"2017-01-05","desc":"dd"}]}

接口描述：

该接口直接查询t\_project\_info表，把所有存在的project信息返回给前端。

### 3、delete\_project

接口名称：删除project信息

前端发送：request:{id: "project\_40"}

返回类型：字符串

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

根据上送的project\_id查询t\_group\_info表，查出所有group，如果没有组，就直接删除t\_project\_info和t\_proj\_seq中数据。如果有组，则判断每一个组的运行状态，任何一个组是运行状态，则直接返回错误，不能删除，否则：

1）先删除所有组件相关的表：T\_DB\_OBJECT\_SET、T\_RAC\_INFO、T\_OBJECT\_SET、T\_COMP\_SEQ、T\_COMP\_PARAM、T\_LOADER\_FILTERS、T\_COMP\_INFO

2）再删除所有组相关的表：T\_GRP\_SEQ、T\_GROUP\_PARAM、T\_COMP\_RELATION、T\_QUEUE\_BPOINT、T\_QUEUE\_PKG、T\_QUEUE\_STATIS、T\_GROUP\_INFO

3）再删除T\_PROJECT\_INFO、T\_PROJ\_SEQ

4）最后删除组相关的目录：data、log、dict、skipsql、errsql、sync、etc

## Graphic类

### 1、add\_graphic

接口名称：保存graphic信息

前端发送：

返回类型：【数组】

接口描述：

Web页面保存组件，前端发起add\_graphic接口，node服务端接收到报文后，

1) 首先处理relationShap信息，relationShap中可能出现的组件信息为capture、appy、transfer，分别进行处理。

capture处理方式为：把capture的source\_db和analysis\_db信息，以及output\_queue信息存放到T\_COMP\_PARAM中，如果T\_COMP\_PARAM中原来有capture的这些信息就进行更新，没有则进行插入。

apply处理方式为：把apply的source\_db、target\_db和input\_queue信息存放到T\_COMP\_PARAM中，如果T\_COMP\_PARAM中原来有apply的这些信息就进行更新，没有则进行插入。

transfer处理方式为：把transfer的source\_db和output\_queue信息存放到T\_COMP\_PARAM中，如果T\_COMP\_PARAM中原来有transfer的这些信息就进行更新，没有则进行插入。

2) 然后再处理line\_object和graphic\_content，直接把line\_object和graphic\_content由json对象转成字符串格式，然后保存到T\_COMP\_RELATION中，如果原来表中有此信息则进行更新，没有则进行插入。

3) 最后遍历graphic\_content内容，把每一个组件信息保存到T\_COMP\_INFO中，如果表中有此组件信息则进行更新，没有则进行插入。

### 2、require\_graphic

接口名称：查询graphic信息

前端发送：{md5: "b7c76fb60fae858d98ff5f06ac46ea6a", request: {group: "group\_5"}}

返回类型：【对象】

{"status":true,"md5":"fa23595e46e13a7a3c171e630946b5e5","response":{"graphic":{"global\_index":"4","line\_object":"[{\"ConnectionId\":\"con\_5\",\"PageSourceId\":\"database3\",\"PageTargetId\":\"capture4\",\"Connector\":\"Flowchart\"}]","graphic\_content":"[{\"BlockId\":\"database3\",\"BlockClass\":\"km-btn-full database-bg drag-cir\",\"BlockTxt\":\"dbwsx\",\"BlockX\":\"153px\",\"BlockY\":\"98px\",\"BlockWidth\":\"94px\",\"BlockHeight\":\"96px\",\"ParentId\":\"flowbox\",\"Rtype\":\"database\",\"Type\":\"oracle\",\"Name\":\"dbwsx\",\"OriginalTitle\":\"undefined\",\"Class\":\"component database jsplumb-draggable jsplumb-droppable \_jsPlumb\_connected \_jsPlumb\_endpoint\_anchor\",\"RealId\":\"database\_6\"},{\"BlockId\":\"capture4\",\"BlockClass\":\"km-btn-full capture-bg drag-cir\",\"BlockTxt\":\"unnamed\",\"BlockX\":\"372px\",\"BlockY\":\"72px\",\"BlockWidth\":\"92px\",\"BlockHeight\":\"95px\",\"ParentId\":\"flowbox\",\"Rtype\":\"capture\",\"Type\":\"undefined\",\"Name\":\"unnamed\",\"OriginalTitle\":\"undefined\",\"Class\":\"component capture jsplumb-draggable jsplumb-droppable \_jsPlumb\_endpoint\_anchor \_jsPlumb\_connected\",\"RealId\":\"capture\_24\"}]"}}}

接口描述：

Web发起require\_graphic接口时， node服务端接收到报文后，在T\_COMP\_RELATION表中根据group\_id查出此组的graphic信息，返回给前端，没有返回空。

## Database类

### 1、save\_db\_info

接口名称：新建database

前端发送：

{"md5":"538dfeada2719bd3e1b0b3504c36553a","request":{"group":"group\_151","component\_name":"db1","db\_type":"oracle","db\_ip":"172.16.6.65","db\_port":"1521","db\_user":"r7","db\_password":"r7","db\_id":"orcl","as\_source\_db":"yes"}}

返回类型：{对象}，把component\_id返回

{"status":true,"md5":"89dd5cbe56a6eece12f704b8b85b1886","response":{"component\_id":"database\_554"}}

接口描述：

输入完元素后，点击保存，前端发起save\_db\_info接口，node服务端接收到报文后，在T\_COMP\_INFO表中插入db组件信息，然后把该db相关的属性(包括db界面显示的名字)插入到表T\_COMP\_PARAM中。最后，把database\_id返给前端。

### 2、query\_db\_info

接口名称：查询database信息

前端发送：

{md5: "ab419f237b782d34c38d687a1a9ac999", request: {group: "group\_5", component\_id: "database\_6"}}

返回类型【数组】：

{"status":true,"md5":"ed5723b85f26044ddcc820556a987aec","response":{"group":"group\_5","component\_id":"database\_6","component\_name":"dbwsx","db\_type":"oracle","db\_ip":"172.16.6.65","db\_port":"1521","db\_user":"r7","db\_password":"r7","db\_id":"orcl","as\_source\_db":"yes"}}

接口描述：

输入完元素后，点击保存，前端发起query\_db\_info接口，node服务端接收到报文后，在T\_COMP\_PARAM表中根据component \_id查出db信息，返回给前端。

### 3、query\_fav\_db

接口名称：查询fav db模板信息

前端发送：{md5: "99914b932bd37a50b983c5e7c90ae93b", request: {}}

返回类型：【对象】。

{"status":true,"md5":"8d20771e9436294ee90c141324620a71","response":{"fav\_dbs":{"db":[{"db\_name":"mb1","db\_type":"oracle","db\_ip":"172.16.6.65","db\_port":"1521","db\_user":"r7","db\_password":"r7","db\_id":"orcl","as\_source\_db":"yes"}]}}}

接口描述：

前端发起query\_fav\_db接口，node服务端接收到报文后，在T\_FAV\_DB表中查出所有db模板信息，返回给前端。

### 4、delete\_fav\_db

接口名称：删除fav db信息

前端发送：{"db\_name":"dd1"}

返回类型：“字符串”：

{"status":true,"md5":"20ebc4141cd715f3bd2851313988792c","response":"SUCCESS"}

接口描述：

前端发起delete\_fav\_db接口，node服务端接收到报文后，在T\_FAV\_DB表中，根据上送的DB\_NAME删除db模板信息，成功返回给前端”SUCCESS”。

### 5、add\_fav\_db

接口名称：增加fav db信息

前端发送：

{"db\_name":"dd","db\_type":"oracle","db\_ip":"172.16.1.225","db\_port":"1521","db\_user":"scott","db\_password":"tiger","db\_id":"orcl","as\_source\_db":"yes"}

返回类型："字符串"：

{"status":true,"md5":"20ebc4141cd715f3bd2851313988792c","response":"SUCCESS"}

接口描述：

输入完元素后，点击保存，前端发起add\_fav\_db接口，node服务端接收到报文后，首先在T\_FAV\_DB表中根据上送的DB\_NAME查询fav db信息是否存在，如果存在返回前端”fav db info is exist”。不存在就插入fav db信息，返回前端"SUCCESS"。

### 6、test\_db\_connection

接口名称：测试db连接

前端发送：

{"db\_type":"oracle","db\_ip":"172.16.1.225","db\_port":"1521","db\_user":"scott","db\_password":"tiger","db\_id":"orcl"}

返回类型："字符串"：

{"status":true,"md5":"20ebc4141cd715f3bd2851313988792c","response":"SUCCESS"}

接口描述：

输入完元素后，点击保存，前端发起test\_db\_connection接口，node服务端接收到报文后，根据不同数据库类型，组装连接串，然后使用用户名和密码去连接数据的相应实例，如果连接成功，返回前端"SUCCESS"，否则返回相应的数据库错误。

### 7、check\_sourcedb\_env

接口名称：检查源端数据库环境

前端发送：

{"db\_is\_source:”yes","db\_type":"oracle","db\_ip":"172.16.1.225","db\_port":"1521","db\_user":"scott","db\_password":"tiger","db\_id":"orcl"}

返回类型：{对象}：

{"archive\_mode":{"status":"success","message":"ARCHIVELOG"},"version":{"status":"success","message":"11.2.0.1.0"},"user\_priv":{"status":"success","message":""},"dbms\_priv":{"status":"success","message":""},"spmt\_log":{"status":"success","message":""}}

接口描述：

输入完元素后，点击保存，前端发起check\_sourcedb\_env接口，node服务端接收到报文后，根据不同数据库类型,进行相应操作”。

oracle:

归档日志模式select log\_mode from v$database。

数据库版本select version from v$instance where rownum<2

用户权限检查

select privilege from user\_sys\_privs

select granted\_role from user\_role\_privs

select table\_name from user\_tab\_privs

DBMS权限检查

select dbms\_flashback.get\_system\_change\_number from dual

select object\_name from all\_objects where object\_name='DBMS\_LOGMNR' and object\_type='PACKAGE'

如果是源端库，检查补充日志

select SUPPLEMENTAL\_LOG\_DATA\_MIN,SUPPLEMENTAL\_LOG\_DATA\_PK,SUPPLEMENTAL\_LOG\_DATA\_UI from v$database

## Capture组件

### 1、save\_parameter

接口名称：保存capture参数

前端发送：

"group": "grp1",

"component\_name": "cap",

"parameter": {

"downstream": "no",

"nls\_lang": "AMERICAN\_AMERICA.ZHS16GBK",

"capture\_interval": "15",

"capture\_heartbeat": "400",

"logminer\_restart\_time": "3009",

"back\_scn": "10",

"transaction\_slot\_size": "32K",

"internal\_queue\_size": "32M",

"h\_send\_sort\_scn": "no",

"scn\_evaluate\_time": "20",

"max\_transaction\_slot": "1000",

"dup\_high\_scn": "200",

"lob\_skip": "no",

"concurrent": "no",

"parallel": "undefined",

"max\_logs": "undefined"

}

返回类型：{对象}：把后台id返回给前端

{"status":true,"md5":"20ebc4141cd715f3bd2851313988792c","response":{"component\_id":"capture\_id"}}

接口描述：

输入完元素后，点击保存，前端发起save\_parameter接口，node服务端接收到报文后，判断capture\_id是否存在t\_comp\_param中，如果存在则进行修改，不存在插入，最后把capture\_id返回给前端。

### 2、add\_capture\_property

接口名称：保存capture

前端发送：

{

"group": "grp1",

"component\_id": "",

"component\_name": "capture",

"db\_component\_name": "db1",

"type": "oracle",

"selected\_users": {

"user": [

"SCOTT",

"WSX"

]

},

"parameter": {

"downstream": "no",

"nls\_lang": "AMERICAN\_AMERICA.ZHS16GBK",

"capture\_interval": "15",

"capture\_heartbeat": "300",

"logminer\_restart\_time": "300",

"back\_scn": "10",

"transaction\_slot\_size": "32K",

"internal\_queue\_size": "32M",

"h\_send\_sort\_scn": "no",

"scn\_evaluate\_time": "20",

"max\_transaction\_slot": "1000",

"dup\_high\_scn": "200",

"lob\_skip": "no",

"concurrent": "no",

"parallel": "undefined",

"max\_logs": "undefined"

}

}

返回类型：{对象}：把component\_id返回给前端

{"status":true,"md5":"20ebc4141cd715f3bd2851313988792c","response":{"component\_id":"capture\_id"}}

接口描述：

点击保存，前端发起add\_capture接口，node服务端接收到报文后，解析参数，判断参数是否在t\_comp\_param中存在，如不存在则进行插入，存在则进行更新。

如果选中了对象，则先把原来t\_comp\_param、t\_object\_set、t\_db\_object\_set中的数据删除，再重新插入。

如果RAC信息存在，则先把原来t\_comp\_param、t\_object\_set、t\_rac\_info中的数据删除，再重新插入。

把params插入数据库中。

### 3、query\_capture\_config

接口名称：查询capture配置

前端发送：

{"md5":"48b8ef6bb964f0bbf9132dcea9698aeb","request":{"group\_id":"group\_150","db\_component\_id":"database\_547","component\_id":"capture\_548","type":"oracle","component\_name":"cap"}}

返回类型：{对象}：

{"status":true,"md5":"a37aa68167e3615b01c6a160b52eb528","response":{"all\_users":{"user":["ANONYMOUS","APEX\_030200","APEX\_PUBLIC\_USER","APPQOSSYS","BI","CC","CTXSYS","DBSNMP","DD","DIP","EXFSYS","FLOWS\_FILES","HR","IX","KYLIN","MDDATA","MDSYS","MGMT\_VIEW","OE","OLAPSYS","ORACLE\_OCM","ORDDATA","ORDPLUGINS","ORDSYS","OUTLN","OWBSYS","OWBSYS\_AUDIT","PM","R7","R9","SCOTT","SH","SI\_INFORMTN\_SCHEMA","SPATIAL\_CSW\_ADMIN\_USR","SPATIAL\_WFS\_ADMIN\_USR","SYS","SYSMAN","SYSTEM","TEST2","WMSYS","XDB","XS$NULL"]},"selected\_users":{"user":["ANONYMOUS","APEX\_030200","APEX\_PUBLIC\_USER","APPQOSSYS","BI","CTXSYS"]},"parameter":{"source\_db":"database\_547","analysis\_db":"database\_547","output\_queue":"queue\_552","downstream":"no","nls\_lang":"AMERICAN\_AMERICA.AL32UTF8","capture\_interval":"15","capture\_heartbeat":"300","logminer\_restart\_time":"300","back\_scn":"10","transaction\_slot\_size":"32K","internal\_queue\_size":"32M","h\_send\_sort\_scn":"no","scn\_evaluate\_time":"20","max\_transaction\_slot":"1000","dup\_high\_scn":"200","lob\_skip":"no","concurrent":"no","parallel":"1","max\_logs":"1","full\_column\_update":"no","checkpoint\_interval":"2h","snapshot\_interval":"10h","exclude\_sql\_text":"","instances":"0"},"rac\_info":{"instance":[]}}}

接口描述：

点击保存，前端发起query\_capture\_config接口，node服务端接收到报文后：

1）查询源端所有用户：select username from dba\_users order by username

2）查询选中对象：如果第一次新建的capture，名字为空，返回选中对象为空。否则从T\_DB\_OBJECT\_SET表中查询出选中对象返回。

3）查询参数：如果第一次新建的capture，名字为空，默认返回相应参数，否则从t\_comp\_param表中查询相应的参数。

4）查询RAC信息：如果第一次新建的capture，名字为空，查询源端数据库RAC信息，否则从T\_RAC\_INFO表中查询相关信息。

## Queue组件

### 1、save\_queue\_info

接口名称：保存queue信息

前端发送：

{"md5":"4f83f86805418800b9e7eec1c41fa4e6","request":{"group\_id":"group\_152","component\_name":"qu1","queue\_type":"file","size":128,"statis":"yes","queue\_save\_hour":24,"component\_id":"queue\_563"}}

返回类型：{对象}：把component\_id返回给前端

{"status":true,"md5":"a96acc7e45ee1b1219f0280ac2f16676","response":{"component\_id":"queue\_563"}}

接口描述：

输入完元素后，点击保存，前端发起save\_queue\_info接口，node服务端接收到报文后，判断component\_id是否存在t\_comp\_info和t\_comp\_param中，如果存在则进行相关信息修改，不存在则插入相关的信息，最后把component\_id返回给前端。

### 2、query\_queue\_info

接口名称：查询queue信息

前端发送：

request:{group\_id: "group\_152", component\_id: "queue\_563"}

返回类型：{对象}：queue信息返回前端

{"status":true,"md5":"42df173fab91cc423b07c477de5c1f68","response":{"group":"group\_152","component\_name":"qu1","group\_id":"group\_152","queue\_type":"file","size":"128M","statis":"yes","queue\_save\_hour":"24"}}

接口描述：

点击queue组件，前端发起query\_queue\_info接口，node服务端接收到报文后，根据component\_id从T\_COMP\_PARAM表中查出信息返回给前端。

### 3、delete\_queue\_pkg\_statis

接口名称：删除队列统计信息

前端发送：group\_id和component\_id。

{"md5":"4deb6332b59fbeee3d70e1f1e45c4a3d","request":{"group\_id":"group\_150","component\_id":"queue\_552","end\_time":"2017-01-23 09:31","flag":1}}

返回类型"字符串"：

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

点击删除队列统计按钮，前端发起delete\_queue\_pkg\_statis接口，上送队列id、起始结束时间，flag标志，flag为1删除T\_QUEUE\_PKG表上送时间段内的数据，flag为2删除T\_QUEUE\_STATIS

表上送时间段内的数据，flag为3，两表都删除上送时间段内的数据。

### 4、query\_queue\_statis

接口名称：获取queue统计信息（注：此接口与原来保持一致）

前端发送：

<dip\_command>

<command>query\_queue\_statis</command>

<command\_data>

<group>group\_1</group>

<component\_name>queue\_5</component\_name>

<full>yes</full>

<owner></owner>

<table></table>

<start\_time>2007-02-1014:38:00</start\_time>

<end\_time>2017-02-1014:38:49</end\_time>

<minute>no</minute>

<hour>no</hour>

<day>no</day>

<month>no</month>

<year>yes</year>

<object\_type></object\_type>

<sql\_type></sql\_type>

<interval>1</interval>

</command\_data>

</dip\_command>

注：group标签对应的值为组ID，component\_name标签对应的值为队列ID

返回类型：{对象}

查询到记录时返回：

{"command\_return":"SUCCESS","return\_data":{"list":{"time":"2016-12","ddl":0,"inst":4,"del":2,"upd":0,"full":6,"length":1498},"persent":{"inst":"66.67%","upd":"0.00%","del":"33.33%","ddl":"0.00%"}}}

未查询到记录时返回：

{"command\_return":"SUCCESS","return\_data":{}}

接口描述：

根据XML中标签值检索t\_queue\_statis表，将检索结果以json的格式返回给前台

### 5、query\_queue\_pkg

接口名称：查询queue事务列表信息

前端发送：

<dip\_command>

<command>query\_queue\_pkg</command>

<command\_data>

<group>tet</group>

<component\_name>li</component\_name>

<page\_num>0</page\_num>

<start\_time></start\_time>

<end\_time></end\_time>

<xid></xid>

</command\_data>

</dip\_command>

注： component\_name标签对应的值为队列ID

返回类型：{对象}

查询到记录时返回：

{"command\_return":"SUCCESS","return\_data":{"list":[{"xid":"10977644350810383","time":"2017-01-1917:42:28","nrec":"1"},{"xid":"9007332398737528","time":"2017-01-1917:39:28","nrec":"1"},{"xid":"1688879925125712","time":"2017-01-1917:34:57","nrec":"1"},{"xid":"6192496732286643","time":"2017-01-1917:34:33","nrec":"1"}],"page\_num":"1"}}

未查询到记录时返回：

{"command\_return":"SUCCESS","return\_data":{"page\_num":0}}

接口描述：

根据XML中标签值检索t\_queue\_pkg表，将检索结果以json的格式返回给前台

### 6、delete\_queue\_pkg\_xid

接口名称：清除指定错误

前端发送：

<dip\_command>

<command>delete\_queue\_pkg\_xid</command>

<command\_data>

<group>group\_1</group>

<component\_name>queue\_4</component\_name>

<xid>2533279085372938</xid>

</command\_data>

</dip\_command>

返回类型：{对象}

{"command\_return":"SUCCESS","return\_data":{"page\_num":0}}

接口描述：

参XML中标签值为条件，删除t\_queue\_pkg表中相应记录，并将操作结果以json的格式返回给前台

## etl组件

### 1、query\_etl\_config

接口名称：查询etl属性信息

前端发送：

{"md5":"51bad4b3230266aab585bb66ac717370","request":{"group":"group\_1","component\_name":"etl","db\_type":"oracle","db\_component\_name":"database\_2","component\_id":"etl\_6"}}

返回类型：{对象}：

{"status":true,"md5":"fb03cf8b9f355e328fc3036fd408807a","response":{"cset":{"real\_charset":"AL32UTF8","charset":["AL32UTF8"]},"ncset":{"real\_ncharset":"AL16UTF16","ncharset":["AL16UTF16"]},"cbset":{"real\_clobset":"AL16UTF16","clobset":["AL16UTF16"]},"ncbset":{"real\_nclobset":"AL16UTF16","nclobset":["AL16UTF16"]},"tset":{"real\_tar\_set":"","tar\_set":["ZHS16GBK","ASCII","GB18030","BIG5","GB2312","UTF-8","UTF-16BE","UTF-16LE","UTF8","UTF16","GBK","US7ASCII","ZHT16BIG5","ZHS16CGB231280","AL32UTF8","AL16UTF16","AL16UTF16LE","WE8MSWIN1252","CP1252","UTF-16","UTF-32","UTF-7","UTF-32BE","UTF-32LE","UTF7","UTF32","UTF16BE","UTF16LE","UTF32BE","UTF32LE","UCS-2","UCS-2LE","UCS-2BE","UCS2"]},"interval":"2","full\_string":"no"}}

接口描述：

双击etl组件，如果源端是oracle，从源端数据库里实时查出charset、ncharset等参数返回给前端。如果是其他类型数据库，默认值返回。

### 2、query\_etl\_user

接口名称：查询etl用户信息

前端发送：

{"md5":"1a9028c9d2249906fde7d8204b97c20b","request":{"group":"group\_6","db\_type":"oracle","db\_component\_name":"database\_24"}}

返回类型：{对象}：etl用户信息返回前端

{"status":true,"md5":"238e5d8e7c4e0fe74df0eedd76e57f99","response":{"user":["ANONYMOUS","APEX\_030200","APEX\_PUBLIC\_USER","APPQOSSYS","BI","CC","CTXSYS","DBSNMP","DD","DIP","DXL","EXFSYS","FLOWS\_FILES","HR","IX","KYLIN","KYLIND","MDDATA","MDSYS","MGMT\_VIEW","OE","OLAPSYS","ORACLE\_OCM","ORDDATA","ORDPLUGINS","ORDSYS","OUTLN","OWBSYS","OWBSYS\_AUDIT","PM","R10","R11","R7","R8","R9","SCOTT","SH","SI\_INFORMTN\_SCHEMA","SPATIAL\_CSW\_ADMIN\_USR","SPATIAL\_WFS\_ADMIN\_USR","SSS","SYS","SYSMAN","SYSTEM","TEST1","U1","WMSYS","XDB","XS$NULL"]}}

接口描述：

前端发起query\_etl\_user接口，根据不同数据类型拼接不同的sql语句，查出所有用户返回给前端。

### 3、query\_etl\_table

接口名称：查询某个用户下的表

前端发送：

{"md5":"66ae96594e6257a6386332644d4df9e6","request":{"group":"group\_6","db\_type":"oracle","db\_component\_name":"database\_24","user":"SCOTT"}}

返回类型"对象"：

{"status":true,"md5":"dbae1bfa063dc94ff8e5adc0183d5f5e","response":{"tables":["SALGRADE","EMP","DEPT","BONUS"]}}

接口描述：

点击某个用户名称，根据不同数据库类型查询出相应用户下的表信息返回给前端。

### 4、query\_etl\_table\_rule

接口名称：查询某个表所配置的规则

前端发送：

{"md5":"2b718a1c5cecf1f51b43d298f373caa1","request":{"group":"group\_6","component\_name":"etl\_28","user":"SCOTT","table":"DEPT"}}

返回类型：{数组}

{"status":true,"md5":"512bc11f03b9acaa90ea1b70dd5aa4d7","response":["add\_column","delete\_column","record\_filter","column\_mapping","table\_audit","operation\_transform"]}

接口描述：

根据用户名、表名，在相应的规则表中查询是否配置规则信息，如果配置，就返回给前端。

### 5、save\_etl\_table\_rule

接口名称：保存某个表所配置的规则

前端发送：

{"md5":"d34f2d7e741a35fce70f8a159e3ad592","request":{"group":"group\_6","component\_name":"etl\_28","user":"SCOTT","table":"DEPT","rules":[{"name":"add\_column","title":"增加列"},{"name":"delete\_column","title":"删除列"},{"name":"record\_filter","title":"记录过滤"},{"name":"column\_mapping","title":"映射列"},{"name":"table\_audit","title":"审计表"},{"name":"operation\_transform","title":"操作转换"}]}}

返回类型：{字符串}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

根据前端所选择的规则信息，把相应的表和规则信息插入到相应的规则表中。

### 6、query\_column\_list

接口名称：查询某个表所有的列信息

前端发送：

{"md5":"2de43c16695b39af843580272e337fed","request":{"group":"group\_6","component\_name":"etl\_28","user":"SCOTT","table":"DEPT","db\_type":"oracle","db\_component\_name":"database\_24","exist\_list":[]}}

返回类型：{对象}

{"status":true,"md5":"380eaa6eec419f85e3560d64e6cf3263","response":{"list":[{"column\_name":"DEPTNO","column\_type":"NUMBER","column\_length":22},{"column\_name":"DNAME","column\_type":"VARCHAR2","column\_length":14},{"column\_name":"LOC","column\_type":"VARCHAR2","column\_length":13}]}}

接口描述：

根据不同的数据库类型，查询选中表的列信息。

### 7、legal\_check\_expression

接口名称：表达式校验

前端发送：

{"md5":"4f2f915560dc3e4b6185601622df99f7","request":{"group":"group\_6","db\_type":"oracle","db\_component\_name":"database\_24","user":"SCOTT","table":"DEPT","connect\_db":"yes","expression":"c3Vic3RyKDpETkFNRSwgMSwgMyk="}}

返回类型：{对象}

{"status":true,"md5":"72e24bc70225ca7d2b8c17692f924d18","response":{"column\_name":["DNAME"]}}

接口描述：

根据上送的表达式，分析出表字段，并把表字段返回给前端。如果源端表中不存在该字段，返回错误。

### 8、query\_etl\_add\_config

接口名称：查询etl配置的增加列规则信息

前端发送：

{"md5":"2b718a1c5cecf1f51b43d298f373caa1","request":{"group":"group\_6","component\_name":"etl\_28","user":"SCOTT","table":"DEPT"}}

返回类型：{对象}

{"status":true,"md5":"d84bba26732b20588a84d1b496cce83f","response":{"database":[{"db\_name":"database\_25","db\_type":"oracle"},{"db\_name":"database\_24","db\_type":"oracle"}],"list":[{"column\_name":"C1","data\_type":"expression","expression":"c3Vic3RyKDpETkFNRSwgMSwgMyk=","connect\_db":"yes","db\_name":"","bind\_name":["DNAME"]}]}}

接口描述：

根据上送的etl、用户、表名，在保存增加列信息的表中查询对应的增加列信息，返回给前端。

### 9、save\_etl\_add\_config

接口名称：保存etl配置的增加列规则信息

前端发送：

{"md5":"539114d41e61a5455db8a2ac6e899a83","request":{"group":"group\_6","component\_name":"etl\_28","db\_type":"oracle","db\_component\_name":"database\_24","user":"SCOTT","table":"DEPT","other\_tables":[],"map\_db\_component\_name":"","map\_db\_type":"","map\_user":"","map\_table":"","list":[{"column\_name":"C1","data\_type":"expression","expression":"c3Vic3RyKDpETkFNRSwgMSwgMyk=","connect\_db":"yes","bind\_name":["DNAME"],"db\_info":{}}]}}

返回类型：{字符串}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

根据前端配置的增加列信息，插入到保存增加列信息的表中。

### 10、query\_etl\_delete\_config

接口名称：查询etl配置的删除列规则信息

前端发送：

{"md5":"2b718a1c5cecf1f51b43d298f373caa1","request":{"group":"group\_6","component\_name":"etl\_28","user":"SCOTT","table":"DEPT"}}

返回类型：{对象}

{"status":true,"md5":"a87050eaa1cd481ed121f7c920460c37","response":{"list":[{"column\_name":"DEPTNO","column\_type":"NUMBER"}]}}

接口描述：

根据上送的etl、用户、表名，在保存删除列信息的表中查询对应的删除列信息，返回给前端。

### 11、save\_etl\_delete\_config

接口名称：保存etl配置的删除列规则信息

前端发送：

{"md5":"f3367ad52c90dd6ef298289cfe7f9d4d","request":{"group":"group\_6","component\_name":"etl\_28","db\_type":"oracle","db\_component\_name":"database\_24","user":"SCOTT","table":"DEPT","other\_tables":[],"list":[{"column\_name":"DEPTNO","column\_type":"NUMBER"}]}}

返回类型：{字符串}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

根据前端配置的删除列信息，插入到保存删除列信息的表中。

### 12、query\_etl\_map\_config

接口名称：查询etl配置的映射列规则信息

前端发送：

{"md5":"2b718a1c5cecf1f51b43d298f373caa1","request":{"group":"group\_6","component\_name":"etl\_28","user":"SCOTT","table":"DEPT"}}

返回类型：{对象}

{"status":true,"md5":"0976b0ea6dbdcaa57dedac8feac972ff","response":{"database":[{"db\_name":"database\_25","db\_type":"oracle"},{"db\_name":"database\_24","db\_type":"oracle"}],"list":[{"column\_name":"DEPTNO","column\_type":"NUMBER","map\_name":"dno","map\_type":"NUMBER"}]}}

接口描述：

根据上送的etl、用户、表名，在保存映射列信息的表中查询对应的映射列信息，返回给前端。

### 13、save\_etl\_map\_config

接口名称：保存etl配置的映射列规则信息

前端发送：

{"md5":"17c069246215735d12f1ea1c672615b0","request":{"group":"group\_6","component\_name":"etl\_28","db\_type":"oracle","db\_component\_name":"database\_24","user":"SCOTT","table":"DEPT","other\_tables":[],"map\_db\_component\_name":"","map\_db\_type":"","map\_user":"","map\_table":"","list":[{"column\_name":"DEPTNO","column\_type":"NUMBER","map\_name":"dno","map\_type":"NUMBER","expression":"","connect\_db":"no","bind\_name":[],"db\_info":{}}]}}

返回类型：{字符串}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

根据前端配置的映射列信息，插入到保存映射列信息的表中。

### 14、query\_etl\_condition\_config

接口名称：查询etl配置的记录过滤规则信息

前端发送：

{"md5":"2b718a1c5cecf1f51b43d298f373caa1","request":{"group":"group\_6","component\_name":"etl\_28","user":"SCOTT","table":"DEPT"}}

返回类型：{对象}

{"status":true,"md5":"4854bc087ac7fb3fb5ad02735d63fc82","response":{"list":[{"expression":"OkROQU1FID0gJ3Rlc3Qn","option":"yes","connect\_db":"yes","db\_name":"","bind\_name":["DNAME"]}]}}

接口描述：

根据上送的etl、用户、表名，在保存记录过滤信息的表中查询对应的记录过滤信息，返回给前端。

### 15、save\_etl\_ condition\_config

接口名称：保存etl配置的记录过滤规则信息

前端发送：

{"md5":"1519e5e8e8b3a6219b2d70199fcccac3","request":{"group":"group\_6","component\_name":"etl\_28","db\_type":"oracle","db\_component\_name":"database\_24","user":"SCOTT","table":"DEPT","other\_tables":[],"list":[{"option":"yes","expression":"OkROQU1FID0gJ3Rlc3Qn","connect\_db":"yes","bind\_name":["DNAME"],"db\_info":{}}]}}

返回类型：{字符串}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

根据前端配置的记录过滤信息，插入到保存记录过滤信息的表中。

### 16、query\_etl\_transform\_config

接口名称：查询etl配置的审计表规则信息

前端发送：

{"md5":"2b718a1c5cecf1f51b43d298f373caa1","request":{"group":"group\_6","component\_name":"etl\_28","user":"SCOTT","table":"DEPT"}}

返回类型：{对象}

{"status":true,"md5":"df6c0d01f07132e5967da52ad72360b3","response":{"database":[{"db\_name":"database\_25","db\_type":"oracle"},{"db\_name":"database\_24","db\_type":"oracle"}],"list":[{"column\_name":"C1","data\_type":"sysdata","column\_value":"scn"}],"table\_prefix":"Pre","table\_suffix":"Suf","keep\_copy":"no"}}

接口描述：

根据上送的etl、用户、表名，在保存审计表信息的表中查询对应的审计表信息，返回给前端。

### 17、save\_etl\_transform\_config

接口名称：保存etl配置的审计表规则信息

前端发送：

{"md5":"f462606921f8b63823a92354243ea635","request":{"group":"group\_6","component\_name":"etl\_28","db\_type":"oracle","db\_component\_name":"database\_24","user":"SCOTT","table":"DEPT","other\_tables":[],"map\_db\_component\_name":"","map\_db\_type":"","map\_user":"","map\_table":"","table\_prefix":"Pre","table\_suffix":"Suf","keep\_copy":"no","list":[{"column\_name":"C1","data\_type":"sysdata","column\_value":"scn"}]}}

返回类型：{字符串}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

根据前端配置的审计表信息，插入到保存审计表信息的表中。

### 18、query\_etl\_convert\_config

接口名称：查询etl配置的操作转换规则信息

前端发送：

{"md5":"c36bd8bc5e5a0593c74cb04721fdc39c","request":{"group":"group\_6","component\_name":"etl\_28","user":"SCOTT","table":"DEPT","oper\_type":"inserttoupdate"}}

返回类型：{对象}

{"status":true,"md5":"abcbff2223f4f66c52d385afab6d9069","response":{"expression":"e0ROQU1FfSA9ICd0ZXN0Jw==","connect\_db":"no","db\_name":"","bind\_name":["DNAME"]}}

接口描述：

根据上送的etl、用户、表名、操作类型，在保存操作转换信息的表中查询对应的操作转换信息，返回给前端。

### 19、save\_etl\_convert\_config

接口名称：保存etl配置的操作转换规则信息

前端发送：

{"md5":"944321fb0d585e755c87f6c9504c87dd","request":{"group":"group\_6","component\_name":"etl\_28","db\_type":"oracle","db\_component\_name":"database\_24","user":"SCOTT","table":"DEPT","operation":{"oper\_type":"inserttoupdate","expression":"e0ROQU1FfSA9ICd0ZXN0Jw==","connect\_db":"no","bind\_name":["DNAME"],"db\_info":{}}}}

返回类型：{字符串}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

根据前端配置的操作转换信息，插入到保存操作转换信息的表中。

### 20、query\_etl\_table\_filter

接口名称：查询etl配置的表过滤规则信息

前端发送：

{"md5":"5f95db88bdae7d00e9eba2d9d69e1b7f","request":{"group":"group\_6","db\_component\_name":"database\_24","db\_type":"oracle","component\_name":"etl\_28","user":"SCOTT"}}返回类型：{对象}

{"status":true,"md5":"3710c399febd22dc29180e506751f761","response":{"flag":"include","tables":["EMP","DEPT"],"list":["SALGRADE","BONUS"]}}

接口描述：

根据上送的etl、用户，在保存表过滤信息的表中查询对应的表过滤信息，返回给前端。

### 21、save\_etl\_table\_filter

接口名称：保存etl配置的表过滤规则信息

前端发送：

{"md5":"0d9ed510222df01476cf1811607c3811","request":{"group":"group\_6","component\_name":"etl\_28","list":[{"user":"SCOTT","flag":"include","list":["EMP","DEPT"]}]}}

返回类型：{字符串}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

根据前端配置的表过滤信息，插入到保存表过滤信息的表中。

### 22、query\_etl\_operation\_filter

接口名称：查询etl配置的操作过滤规则信息

前端发送：

{"md5":"5f95db88bdae7d00e9eba2d9d69e1b7f","request":{"group":"group\_6","db\_component\_name":"database\_24","db\_type":"oracle","component\_name":"etl\_28","user":"SCOTT"}}返回类型：{对象}

{"status":true,"md5":"77009b1a941606feb2f0f3958d5da107","response":{"list":[{"tab\_name":"EMP","tab\_ins":true,"tab\_del":true,"tab\_upt":false},{"tab\_name":"SALGRADE","tab\_ins":false,"tab\_del":false,"tab\_upt":false},{"tab\_name":"DEPT","tab\_ins":false,"tab\_del":false,"tab\_upt":false},{"tab\_name":"BONUS","tab\_ins":false,"tab\_del":false,"tab\_upt":false}]}}

接口描述：

根据上送的etl、用户，在保存操作过滤信息的表中查询对应的操作过滤信息，返回给前端。

### 23、save\_etl\_operation\_filter

接口名称：保存etl配置的操作过滤规则信息

前端发送：

{"md5":"427056147a7bf4f8944aa76550716ab0","request":{"group":"group\_6","db\_component\_name":"database\_24","db\_type":"oracle","component\_name":"etl\_28","list":[{"user":"SCOTT","list":[{"id":1,"name":"EMP","tab\_ins":true,"tab\_del":true,"tab\_upt":false,"expression":"","connect\_db":"no","bdList":[],"dbInfo":{}}]}]}}

返回类型：{字符串}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

根据前端配置的操作过滤信息，插入到保存操作过滤信息的表中。

### 24、query\_batch

接口名称：保存etl配置的操作过滤规则信息

前端发送：

{"md5":"43927bb3807eb6c563e264d86151db46","request":{"group":"group\_6","db\_component\_name":"database\_24","db\_type":"oracle","rule\_type":"filter"}}

返回类型：{数组}

{"status":true,"md5":"26ae65bcfbdd7445019adc8939fc996e","response":{"batch\_name":["filter1"]}}

接口描述：

根据前端发送的组名、规则类型，查询改组下所配置的批量信息。

### 25、save\_batch\_rule

接口名称：保存批量规则信息

前端发送：

{"md5":"34ac894a1fde87d2480f1f2e80595618","request":{"group":"group\_6","db\_component\_name":"database\_24","db\_type":"oracle","rule\_type":"filter","batch\_name":"filter1","rules":[{"id":0,"checked":false,"coder":"CODE1","resered":"yes","expression":"OkROQU1FID0gJXM=","value":":DNAME = %...","bdList":["DNAME"],"dbInfo":{},"list":[{"user":"SCOTT","list":["EMP","DEPT"]}],"connect\_db":"yes"}]}}

返回类型：{字符串}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

根据前端配置的批量规则信息，插入到保存批量信息的表中。

### 26、query\_batch\_rule

接口名称：查询具体批量规则信息

前端发送：

{"md5":"48c6e593b12145c1131f3279a5b75793","request":{"group":"group\_6","db\_component\_name":"database\_24","db\_type":"oracle","rule\_type":"filter","batch\_name":"filter1"}}

返回类型：{字符串}

{"status":true,"md5":"736b57e832dcd9783f4df82684c26241","response":{"rules":[{"id":0,"checked":false,"coder":"CODE1","resered":"yes","expression":"OkROQU1FID0gJXM=","value":":DNAME = %...","bdList":["DNAME"],"dbInfo":{},"list":[{"user":"SCOTT","list":["EMP","DEPT"]}],"connect\_db":"yes"}]}}

接口描述：

根据前端上送的批量规则名称和类型，查询该规则集内具体规则信息。

### 27、query\_batch\_checked

接口名称：查询etl所配置的批量信息

前端发送：

{"md5":"3259be2f93069e6fb4408a951c96b942","request":{"group":"group\_6","db\_component\_name":"database\_24","db\_type":"oracle","component\_name":"etl\_28","rule\_type":"filter"}}

返回类型：{对象}

{"status":true,"md5":"8afeaf8ec1f80593fbb6975370ee93d6","response":{"list":[{"name":"filter1","checked":false,"codes":[{"code":"CODE1","value":""}]}]}}

接口描述：

根据前端上送的批量规则类型，查询已选中和未选中的规则集信息。

### 28、save\_batch\_codevalue

接口名称：保存已选中的批量信息

前端发送：

{"md5":"8debd2affd162ff48349c38a585e185c","request":{"group":"group\_6","db\_component\_name":"database\_24","db\_type":"oracle","component\_name":"etl\_28","rule\_type":"filter","batchRules":[{"name":"filter1","checked":true,"codes":[{"code":"CODE1","value":"'hello'"}]}]}}

返回类型：{字符串}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

根据前端上送的批量规则信息，插入到批量规则信息表中。

### 29、query\_etl\_rule\_ptable

接口名称：查询etl已配置规则的对象信息

前端发送：

{"md5":"a4587a47ae78cf52a38f852118f50213","request":{"group":"group\_6","db\_component\_name":"database\_24","db\_type":"oracle","component\_name":"etl\_28","rule\_type":"add\_column"}}

返回类型：{对象}

{"status":true,"md5":"894c083c174f328ce9ed1936ff1af28e","response":{"list":[{"user":"SCOTT","table":"DEPT"}]}}

接口描述：

根据前端上送的规则类型，查询所配置该类型规则的对象信息。

## Loader组件

### 1、query\_apply\_config

接口名称：查询组件参数

前端发送：

{group\_id: "group\_53", component\_id: "apply\_245", db\_component\_id: "database\_238", db\_type: "oracle"}

返回类型：{对象}

{"status":true,"md5":"c35e5dbd0d3e2f4c4e03dda2a2640839","response":{"objects":{"user":[{"name":"ANONYMOUS","object\_type":[{"name":"INDEX"},{"name":"TABLE"},{"name":"VIEW"},{"name":"SEQUENCE"},{"name":"PROCEDURE"},{"name":"FUNCTION"},{"name":"PACKAGE"},{"name":"PACKAGE\_BODY"},{"name":"TRIGGER"},{"name":"TYPE"},{"name":"TYPE\_BODY"}]},{"name":"APEX\_030200","object\_type":[{"name":"INDEX"},{"name":"TABLE"},{"name":"VIEW"},{"name":"SEQUENCE"},{"name":"PROCEDURE"},{"name":"FUNCTION"},{"name":"PACKAGE"},{"name":"PACKAGE\_BODY"},{"name":"TRIGGER"},{"name":"TYPE"},{"name":"TYPE\_BODY"}]},{"name":"APEX\_PUBLIC\_USER","object\_type":[{"name":"INDEX"},{"name":"TABLE"},{"name":"VIEW"},{"name":"SEQUENCE"},{"name":"PROCEDURE"},{"name":"FUNCTION"},{"name":"PACKAGE"},{"name":"PACKAGE\_BODY"},{"name":"TRIGGER"},{"name":"TYPE"},{"name":"TYPE\_BODY"}]},{"name":"APPQOSSYS","object\_type":[{"name":"INDEX"},{"name":"TABLE"},{"name":"VIEW"},{"name":"SEQUENCE"},{"name":"PROCEDURE"},{"name":"FUNCTION"},{"name":"PACKAGE"},{"name":"PACKAGE\_BODY"},{"name":"TRIGGER"},{"name":"TYPE"},{"name":"TYPE\_BODY"}]}]},"filters":{"filter":[{"filter\_type":"INCLUDE","schema":[]},{"filter\_type":"EXCLUDE","schema":[]}]},"parameter":{"source\_db":"database\_238","target\_db":"database\_246","input\_queue":"queue\_247"}}}

接口描述：

前端点击apply组件发起此接口，目前包括loader，node服务端接收到报文后，对于loader：

1）查询用户信息：如果capture不存在，则查询源端数据库所有对象，否则查询capture选中的对象。

2）查询参数：如果第一次新建，name为空，则返回默认相应参数，否则查询t\_comp\_param表返回参数

3）查询选中的对象：如果第一次新建，name为空，则返回空，否则查询T\_LOADER\_FILTERS表，返回选中的对象。

### 2、add\_apply\_config

接口名称：保存apply参数

前端发送：

{group\_id: "group\_53", db\_component\_id: "database\_238", component\_name: "ds", type: "oracle",…}

返回类型：{对象}：把component\_id返回给前端

{"status":true,"md5":"20ebc4141cd715f3bd2851313988792c","response":{"component\_id":"capture\_id"}}

接口描述：

点击apply保存，前端发起add\_apply\_config接口。

如果是loader：

1）插入T\_COMP\_INFO表，存在则更新。插入T\_LOADER\_FILTERS表，存在则更新。

2）首先删除t\_comp\_param、t\_object\_set、t\_db\_object中存在的信息，解析选中的对象信息，分别插入到t\_comp\_param、t\_object\_set、t\_db\_object\_set表中。

3）把loader的参数插入到表t\_comp\_param中，存在则更新。

4）把选中的对象信息和参数保存到组下面，apply组件xml文件中，供全同步使用。

如果是transfer:

把相关信息插入到t\_comp\_info和t\_comp\_param表中，如果表中存在对应的信息，则删除，重新插入。

### 3、query\_apply\_sourcedb\_object

接口名称：查询源端数据库对象信息

前端发送：

request:{group\_id: "group\_53", type: "oracle", db\_component\_id: "database\_238", object\_type: "INDEX",…}

返回类型：{对象}

{"status":true,"md5":"7680a764dea7a8ff1c2d53b0745dd979","response":{"objects":{"object":[]}}}

接口描述：

点击某一个对象，比如：INDEX，在源端数据库里查询此对象信息返回给前端。

## TClient/TServer传输组件

### 1、query\_qrecv\_config

接口名称：查询tclient组件参数

前端发送：

request:{group\_id: "group\_49", component\_id: "transfer\_231"}

返回类型：{对象}

{"status":true,"md5":"20a0604b72bac4db5f23aef3c7e1779e","response":{"group\_name":"tserver\_test","ip":"172.16.6.204","port":"7007","compress":"yes","encrypt":"yes","queue\_name":"queue","pkg\_size":"1024","db\_type":"oracle","user":"admin","passwd":"11111111"}}接口描述：

前端点击tclient组件发起此接口，node服务端接收到报文后，根据component\_id从t\_comp\_param表中查询出tclient参数返回给前端。

### 2、add\_apply\_config

接口名称：保存tclient参数

前端发送：  
{group\_id: "group\_49", component\_name: "tclient", type: "transfer",…}

返回类型：{对象}：把后台id返回给前端

{"status":true,"md5":"e319c602cc97421ca3239773909cded1","response":{"component\_id":"transfer\_231"}}

接口描述：

点击保存，前端发起add\_apply\_config接口，node服务端接收到报文后,把tclient相关信息参数到数据库表t\_comp\_info和t\_comp\_param中。

### 3、query\_tserver\_config

接口名称：查询tserver信息

前端发送：

request:{}

返回类型：{对象}

{"status":true,"md5":"4740a59d1aed99b1b1d2f341bde8d615","response":{"server\_config":{"user":{"name":"wangsx","passwd":"555555555","all\_que":"yes"}}}}

接口描述：

点击tserver，查询t\_server\_param表，返回tserver的用户名、密码等信息。

### 4、add\_tserver\_config

接口名称：增加tserver信息

前端发送：

request:{server\_config: {user: {name: "wangsx", passwd: "555555555", pri: {all\_que: "yes"}}}}

返回类型：{对象}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

配置好tserver用户名和密码，点击保存，把用户名和密码插入到数据库表t\_server\_param中。

## 删除组件delete\_component

接口名称：删除组件信息

前端发送：

request:{group: "group\_49", component\_name: "queue", type: "apply", component\_id: "apply\_235"}

返回类型：{对象}

{"status":true,"md5":"d0749aaba8b833466dfcbb0428e4f89c","response":"SUCCESS"}

接口描述：

选中某一个组件，点击删除，根据不同组件类型，做相应的操作：

首先检查组是否运行，如果组正在运行中，则返回错误，否则：

capture组件：删除T\_DB\_OBJECT\_SET、T\_RAC\_INFO、T\_OBJECT\_SET、T\_COMP\_SEQ、T\_COMP\_PARAM、T\_COMP\_INFO表信息，并删除相应的bkp文件

queue组件：删除T\_QUEUE\_BPOINT、T\_QUEUE\_PKG、T\_QUEUE\_STATIS、T\_COMP\_SEQ、T\_COMP\_PARAM、T\_COMP\_INFO表信息，并删除相应的ctl文件和log文件。

apply组件：删除T\_DB\_OBJECT\_SET、T\_OBJECT\_SET、T\_LOADER\_FILTERS、T\_COMP\_SEQ、T\_COMP\_PARAM、T\_COMP\_INFO表信息，并删除相应的bkp文件和xml配置文件。

transfer组件：直接删除T\_COMP\_SEQ、T\_COMP\_PARAM、T\_COMP\_INFO表信息，并删除相应的bkp文件。

## Group类

### add\_group

接口名称：新建group

前端发送：

{"projectid":'722',"group":'group\_5',"description":'test1',"log\_save\_hour":48,"users":'admin',"auth":'super\_admin'}

注：projectid属性对应值为工程ID

返回类型：{对象}

操作失败返回：

{"status":false,"md5":"0b35b9c2f960fb6be99e8d33e3996f9d","response":{"error\_code":"00002","error\_msg":"00002:is exist"}}

操作成功返回：

{"status":true,"md5":"4bc1b2fd7de221014f2052cfed083eb6","response":{"group\_id":"group\_50"}}

接口描述：

首先，以工程ID和组名为条件查询数据库表t\_group\_info，如果有记录则说明已存在同名的组，此时返回“已存在”的错误；如果没有记录，则检查目前数据库表中已有group的数量是否超出配置文件中配置的最大个数，超出则返回错误，否则插入group及相关参数信息,然后把刚插入的group\_id返回前端。

### del\_group

接口名称：删除group

前端发送：

{"group":'group\_5'}

注：group属性对应的值为组ID

返回类型：{对象}

{"status":true,"md5":"99914b932bd37a50b983c5e7c90ae93b","response":{}}

接口描述：

以前台传来的组ID为条件查询t\_group\_info表，如果有记录存在则删除该组相关的信息（组信息、组参数及与组相关的目录结构）

### modify\_group

接口名称：修改group

前端发送：

{"group":'grp\_name',"new\_group":'new\_name',"description":'proj2\_1226',"log\_save\_hour":48,”group\_id”:’grop\_1’}

注：group属性对应的值为组名，group\_id属性对应的值为组ID

返回类型：{对象}

{"status":true,"md5":"99914b932bd37a50b983c5e7c90ae93b","response":{}}

接口描述：

以前台上传的group\_id为条件更新组信息及组参数

### reset\_group

接口名称：重置group

前端发送：

{"group":'group\_5',"clean\_all":'no'}

注：group属性对应的值为组ID

返回类型：{对象}

操作成功返回：

{"status":true,"md5":"99914b932bd37a50b983c5e7c90ae93b","response":{}}

操作失败返回：

{"status":false,"md5":"8d0632f51135488c0f4d78626778d061","response":{"error\_code":"00001","error\_msg":"00001:not exist"}}

接口描述：

首先，检查组状态，如果组处于运行状态则向前台返回“EBUSY”；如果用户选择只清除断点，则清除断点文件后返回成功；如果用户选择清除断点、队列和日志文件，则删除相关文件后返回成功。

### query\_group

接口名称：查询group

前端发送：

{"projectid":722,"users":'admin',"auth":'super\_admin'}

注：projectid属性对应的值为工程ID

返回类型：[{对象}]

查询到记录时返回：

{"status":true,"md5":"ddb8d4a34693893a8a076475ceaed9ea","response":{"group":[{"name":"group\_5","description":"test1","log\_save\_hour":"96","loaded":"no"},{"name":"group\_6","description":"test1","log\_save\_hour":"48","loaded":"no"}]}}

未查询到记录时返回：

{"status":true,"md5":"f1f1c97b6b45aa74681b79cc95d50247","response":{"group":[]}}

接口描述：

首先，以前台上传的projectid及auth为条件对表t\_group\_info和t\_project\_info时行联合查询，检索出符合条件的组；其次，与manager进行交互查询各组的状态；最后，将组信息及状态信息返回给前台

## System\_setting类

### save\_reporter\_conf

接口名称: 保存系统配置错误检测系统参数

前端发送：

{ md5: "37e3cc9042655c432e93930002352a6a",

request:{ report\_system:

{ task\_identify: 'system',

monitor\_ip: '172.16.1.58',

monitor\_port: '9999',

monitor\_interval: '1m',

free\_disk\_threshold: '300G',

capture\_interval\_threshold: '',

unmonitored\_group: { group: [ 'aa', '', ''] } } }

返回类型：{对象}

{"status":1,"md5":"99914b932bd37a50b983c5e7c90ae93b","response":{}}

接口描述：

首先，查询表t\_service\_info中type为MONITOR的记录对应的id；然后，以该id为条件清除表t\_service\_param中相应记录；最后，将前台上传的json对象插入到t\_service\_param表中

### save\_monitor\_conf

接口名称: 保存系统配置错误发送系统参数

前端发送：

{ md5: "37e3cc9042655c432e93930002352a6a",

request:{ monitor:

{ customer\_name: 'system\_one',

server\_port: '9999',

alert\_type: 'all',

mail:

{ mail\_server: 'smtp.r7data.com',

mail\_port: '25',

user: 'jiangyp@r7data.com',

password: 'Xingqin0',

address\_list: { address: 'jiangyp@r7data.com' } },

sms:

{ send\_method: { method: 'default' },

phone\_number\_list: { phone\_number: '18611801700' } } } }

返回类型：{对象}

{"status":1,"md5":"99914b932bd37a50b983c5e7c90ae93b","response":{}}

接口描述：

首先，查询表t\_service\_info中type为SENDER的记录对应的id；然后，以该id为条件清除表t\_service\_param中相应记录；最后，将前台上传的json对象插入到t\_service\_param表中

### query\_reporter\_conf

接口名称：查询系统配置错误检测系统参数

前端发送：

{md5: "99914b932bd37a50b983c5e7c90ae93b", request: {}}

返回类型：{对象}

{"status":1,"md5":"3aa7c75f14c97f956d2fe80d29616304","response":{"report\_system":{"monitor\_disk\_name":"sda","update\_interval":"300","monitor\_interval":"2","free\_disk\_threshold":"20","unmonitored\_group":{}}}}

接口描述：

首先，查询表t\_service\_info中type为SENDER的记录对应的id；然后，以该id为条件查询表t\_service\_param中相应记录；最后，将查询结果以json的形式传送到前端。

### query\_monitor\_conf

接口名称：查询系统配置错误发送系统参数

前端发送：

{md5: "99914b932bd37a50b983c5e7c90ae93b", request: {}}

返回类型：{对象}

{"status":1,"md5":"8a8917aaa50870010ddadaf27ab371b1","response":{"monitor":{"customer\_name":"test","server\_port":"9999","alert\_type":"all","mail":{"mail\_server":"smtp.r7data.com","mail\_port":"25","user":"zhanghf@r7data.com","password":"loveangel2239","address\_list":{"address":["zhanghf@r7data.com"]}},"sms":{"phone\_number\_list":{"phone\_number":["15652539836"]}}}}}

接口描述：

首先，查询表t\_service\_info中type为MONITOR的记录对应的id；然后，以该id为条件查询表t\_service\_param中相应记录；最后，将查询结果以json的形式传送到前端。

### dip\_manuser\_query\_user

接口名称：查询系统配置用户列表

前端发送：

{md5: "196bd9c7a5e5a4f5eb580f2631078aad", request: {founder: "admin"}}

返回类型：{对象}

{"status":1,"md5":"13172c1850d8860a74dcb48dea8ca411","response":{"list":[{"user":"admin","passwd":"628zqOdd1dY=","authority":"super\_admin"},{"user":"jiangyp","passwd":"1oGrLuvQmdLzmTEDF7uodg==","authority":"super"},{"user":"zhf","passwd":"6Cn1mIrKFVIYJKxlnPY0zw==","authority":"super"}]}}

接口描述：

首先，以前台上送的founder为条件查询t\_user\_manager，检索出由founder创建的用户和founder本身，并将检索结果以json的形式传送到前端

### dip\_manuser\_save\_user

接口名称：系统配置添加新用户

前端发送：

{md5: "703568bd225ccddc16effd9468f16e46",request:{user: "zhf", passwd: "zhf12345", authority: "super", founder: "admin", type: "add"}}

返回类型：{对象}

正确添加时返回结果：

{"status":1,"md5":"99914b932bd37a50b983c5e7c90ae93b","response":{}}

用户已存在时返回结果：

{"status":0,"md5":"0b35b9c2f960fb6be99e8d33e3996f9d","response":{"error\_code":"00002","error\_msg":"00002:is exist"}}

接口描述：

以前台上送的user为条件查询t\_user\_manager，如果用户已存在则返回给前台的结果为“用户已存在”，否则返回添加成功

### dip\_manuser\_save\_user

接口名称：系统配置删除新用户

前端发送：

{md5: "ae6cd66c149cf0a2f576f6c0eec9bd1e"",request: {user: "zhf1", passwd: "", authority: "super", founder: "admin", type: "del"}}

返回类型：{对象}

{"status":1,"md5":"99914b932bd37a50b983c5e7c90ae93b","response":{}}

接口描述：

以前端上送的user、founder值为条件，删除表t\_user\_manager中相关记录，并将执行结果以json形式返回到前端

### dip\_manuser\_save\_user

接口名称：系统配置添加新用户

前端发送：

{md5: "ae6cd66c149cf0a2f576f6c0eec9bd1e"",request: {user: "zhf", passwd: "zhf12345", authority: "super", founder: "admin", type: "alter"}}

返回类型：{对象}

{"status":1,"md5":"99914b932bd37a50b983c5e7c90ae93b","response":{}}

接口描述：

以前端上送的user值为条件，更新表t\_user\_manager中USER\_PWD字段的值，并将执行结果以json形式返回到前端

## Tool\_set类

### export\_config

接口名称：配置备份

前端发送：

{md5: "99914b932bd37a50b983c5e7c90ae93b", request: {}}

返回类型：{对象}

{"status":1,"md5":"b1f26c2892c6ad492ee246a08dffb971","response":{"filename":"dip\_etc\_20170213103248.tar.gz"}}

接口描述：

首先，将配置相关表的数据导出到以表名命名的脚本文件中；其次，将$DIP\_HOME/etc目录下相关内容打包；再次，将前两步生成的内容打包并压缩； 最后，将最终的压缩文件名以json格式发送到前端

注：

1. 配置相关表：T\_USER\_MANAGER、T\_PROJ\_SEQ、T\_GRP\_SEQ、T\_COMP\_SEQ、T\_PROJECT\_INFO、T\_GROUP\_INFO、T\_GROUP\_PARAM、T\_SERVICE\_INFO、T\_SERVICE\_PARAM、T\_COMP\_INFO、T\_COMP\_PARAM、T\_COMP\_RELATION、T\_FAV\_DB、T\_OBJECT\_SET、T\_DB\_OBJECT\_SET、T\_LOADER\_FILTERS、T\_RAC\_INFO

2. 压缩文件名的命名格式：dip\_etc\_”YYYYMMDDmmss”.tar.gz

### import\_config

接口名称：配置还原

前端发送：

{md5:'edfadfac3a554347e2d4fc0481107720',request:{filename:’dip\_etc\_20170209192238.tar.gz' } }

返回类型：{对象}

成功返回：

{ status: 1,md5: '99914b932bd37a50b983c5e7c90ae93b',response: {} }

有组运行时返回：

{"status":0,"md5":"3a85a5557d34b47899e8536ca07cc54d","response":{"error\_code":"00010","error\_msg":"00010:EBUSY"}}

权限不足时返回：

{"status":0,"md5":"19cdb2a52b0eba50ee59948dca372160","response":{"error\_code":"00017","error\_msg":"00017:Permission Denied"}}

接口描述：

首先，检查是否有组正在运行，如果有组在运行则返回错误；其次，检查当前登录用户是否为超级管理员，如果不为超级管理员则返回错误； 最后，如果没有组在运行且当前登录用户为超级用户则将前端上传的包解压并执行相关脚本，脚本执行完后返回。

### export\_errinfo

接口名称：导出错误

前端发送：

{md5: "99914b932bd37a50b983c5e7c90ae93b", request: {}}

返回类型：{对象}

{"status":1,"md5":"7ab477349ece8a7370a417cfe75cae3a","response":{"filename":"dip\_errinfo\_20170213112343.tar.gz"}}

接口描述：

首先，将$DIP\_HOME/errinfo目录下相关内容打包，并压缩；其次，将上步中的压缩包移到$DIP\_HOME/ web/app/download/目录 下； 最后，将最终的压缩文件名以json格式发送到前端

注：压缩文件名的命名格式：dip\_errinfo\_”YYYYMMDDmmss”.tar.gz

## Error类

### query\_apply\_full\_error

接口名称：获取Loader 错误列表

前端发送：

<dip\_command>

<command>query\_apply\_full\_error</command>

<command\_data>

<group>group\_1</group>

<component\_name>apply\_6</component\_name>

<owner></owner>

<object></object>

<action></action>

<page\_num>0</page\_num>

<mark\_flag>0</mark\_flag>

<start\_time></start\_time>

<end\_time></end\_time>

</command\_data>

</dip\_command>

返回类型：{对象}

{"command\_return":"SUCCESS","return\_data":{"all\_num":1,"history":[{"detailed\_message":"SCN=[17419105] XID=[1125912791754350]\nROWID=[AAATSMAAEAAAWOdAAA]\n\nError Code: 942\nError Message: ORA-00942: table or view does not exist\n\n\nERROR SQL:\nINSERT INTO \"R7\".\"test\_zhf\"(\n\"ID\",\n\"NAME\",\n\"NAME2\",\n\"NAME3\"\n)\nVALUES\n(\n 1,\n'dsadad',\n'dasd',\n'sdaad'\n)\n","time":"2017-01-24T02:04:11.000Z","flag":"0","owner":"R7","object":"TEST1","action":"UnknownAction","offset":"","operation":"INSERT","error\_type":"Access Database Error942"}]}}

接口描述：

首先，以前端发送的值为条件检索错误索引表（t\_ldr\_err\_idx），得出错误在文件中的偏移和长度；其次，从错误文件（”compnentid”\_error\_sql.log）中读出错误信息；最后，将结果以json的形式返回到前端

### query\_apply\_current\_error

接口名称：获取Loader 当前错误

前端发送：

<dip\_command>

<command>query\_apply\_current\_error</command>

<command\_data>

<group>group\_1</group>

<component\_name>apply\_5</component\_name>

</command\_data>

</dip\_command>

返回类型：{对象}

{"command\_return":"SUCCESS","return\_data":{"time":"2017-01-24T02:04:11","owner":"R7","object":"TEST1","status":"Running","detailed\_message":"SCN=[17419105]XID=[1125912791754350]\nROWID=[AAATSMAAEAAAWOdAAA]\n\nError Code: 942\nError Message: ORA-00942: table or view does not exist\n\n\nERROR SQL:\nINSERT INTO \"R7\".\"test\_zhf\" (\n \"ID\",\n \"NAME\",\n \"NAME2\",\n \"NAME3\"\n)\nVALUES\n(\n 1,\n 'dsadad',\n 'dasd',\n 'sdaad'\n)\n"}}

接口描述：

首先，从错误索引表（t\_ldr\_err\_idx）中检索出最后一条错误对应的信息；其次，从错误文件（”compnentid”\_error\_sql.log）中读出错误信息；最后，将结果以json的形式返回到前端

### trunc\_apply\_error\_record

接口名称：清除错误日志

前端发送：

<dip\_command>

<command>trunc\_apply\_error\_record</command>

<command\_data>

<group>tet</group>

<component\_name>lo</component\_name>

</command\_data>

</dip\_command>

返回类型：{对象}

{"command\_return":"SUCCESS","return\_data":{}}

接口描述：

首先，truncate错误索引表（t\_ldr\_err\_idx）；其次，truncate错误文件（lo\_error\_sql.log）；最后，将结执行果以json的形式返回到前端

注：

错误文件命名格式： “组件ID”\_error\_sql.log

### marked\_apply\_error

接口名称：将指定错误标记为已读

前端发送：

<dip\_command>

<command>marked\_apply\_error</command>

<command\_data>

<group>group\_1</group>

<component\_name>apply\_7</component\_name>

<location>

<offset>0</offset>

<offset>10</offset>

</location>

</command\_data>

</dip\_command>

返回类型：{对象}

{"command\_return":"SUCCESS"}

接口描述：

首先，从前端上传的xml中解析出组ID、组件ID及错误信息的偏移；其次，以上一步中解析出的值为条件将错误索引表（t\_ldr\_err\_idx）中相关记录的falg字段设置为1；最后，将结执行果以json的形式返回到前端

### download\_error\_file

接口名称：将历史错误导出为Excel

前端发送：

<dip\_command>

<command>download\_error\_file</command>

<command\_data>

<group>tet</group>

<component\_name>lo</component\_name>

</command\_data>

</dip\_command>

返回类型：{对象}

没有错误时返回：

{"command\_return":"ERROR","return\_data":{"error\_message":"no errors"}}

存在错误时返回：

{"command\_return":"SUCCESS","return\_data":{"file\_path":"group\_1\_apply\_6\_error\_sql.csv"}}

接口描述：

首先，从前端上传的xml中解析出组ID和组件ID；其次，以组ID和组件ID为条件检索错误索引表（t\_ldr\_err\_idx）；再次，将检索结果生成csv格式文件；最后，将csv文件名返回给前端

注：

csv文件名格式: “组ID”\_”组件ID”\_error\_sql.csv